DETERMINING CARGO FLOW FOR AIR MOBILITY COMMAND'S CHANNEL CARGO SYSTEM

THESIS

Michael Del Rosario, Captain, U.S. Army

AFIT/GOR/ENS/93M-04

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GOR/ENS/93M-04

DETERMINING CARGO FLOW
FOR AIR MOBILITY COMMAND'S
CHANNEL CARGO SYSTEM

THESIS

Michael Del Rosario, Captain, U.S. Army

AFIT/GOR/ENS/93M-04

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)

2. REPORT DATE
March 1993

3. REPORT TYPE AND DATES COVERED
Master's Thesis

4. TITLE AND SUBTITLE
DETERMINING CARGO FLOW FOR AIR MOBILITY COMMAND'S CHANNEL CARGO SYSTEM

5. FUNDING NUMBERS

6. AUTHOR(S)
Michael Del Rosario, CPT, U.S. Army

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology,
WPAFB, OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GOR/ENS/93M-04

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
Approved for public release; distribution unlimited

12a. DISTRIBUTION/AVAILABILITY STATEMENT

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)
This research investigated using a multiperiod multicommodity minimal cost flow (M2MCF) formulation to model the channel cargo system of the United States Air Force's Air Mobility Command (AMC). The objective of this research was to determine how cargo should flow in the channel cargo system in order to minimize the cargo's delay enroute from its origin to its destination. This research showed the size of an M2MCF model of the channel cargo system exceeds AMC's computational capabilities. Additionally, this research describes three approaches to reduce the problem size. Because of the problem size and other modeling limitations discovered during this research, the presented M2MCF model of the channel cargo system is currently not accurate enough to be useful as a scheduling tool. However, the M2MCF model may be adequate for AMC advance planning purposes. Furthermore, the M2MCF model dual variables may yield useful information for the improvement of AMC's monthly flight schedule. Finally, this research recommends ways to eliminate the limitations associated with the M2MCF model.

14. SUBJECT TERMS
Network Flows, Logistics, Multiperiod, Multicommodity Minimal Cost Flow, Dual Variables, Linear Programming

15. NUMBER OF PAGES
195

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std Z39-18
298-102

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

AFIT/GOR/ENS/93M-04

DETERMINING CARGO FLOW

FOR AIR MOBILITY COMMAND'S

CHANNEL CARGO SYSTEM

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Operations Research

Michael Del Rosario, B.S., P.E.

Captain, U.S. Army

March 1993

Thesis Approval

STUDENT: CPT Michael Del Rosario CLASS: GOR-93M

THESIS TITLE: Determining Cargo Flow for Air Mobility Command's Channel Cargo System

DEFENSE DATE: 26 February 1993

COMMITTEE:	NAME/DEPARTMENT	SIGNATURE
Advisor:	MAJ John J. Borsi/ENS	John J. Borsi
Reader:	LTC James T. Moore/ENS	James T. Moore

Acknowledgements

I owe many people my thanks for their assistance in this research. First of all, I wish to thank MAJ John J. Borsi and LTC James T. Moore for serving not only as advisor and reader but also as educators, counselors and mentors. I truly appreciate their patience and their willingness to share their vast knowledge and experience. I also must thank several people at the AMC Force Structure Analysis Office: LTC Joseph Litko, Mr. Alan Whisman, 1LT Jonathan Robinson, and MAJ Reed Hanson. Their cooperation and immediate responses to my requests are sincerely appreciated. I am also indebted to CPT Greg Rau, CPT Rich Pace, and Mr. Erwin Kalvelagen whose assistance in FORTRAN and GAMS programming were invaluable. Last and most important, I must thank my wife, Teresa, and our three children, Gregory, Michelle, and David, who gave me support while I served as a full-time student and only a part-time husband and father. I cannot fully express my gratitude to them.

Michael Del Rosario

Table of Contents

List of Figures

List of Tables

AFIT/GOR/ENS/93M-04

Abstract

This research investigated using a multiperiod multicommodity minimal cost flow (M^2MCF) formulation to model the channel cargo system of the United States Air Force's Air Mobility Command (AMC). The objective of this research was to determine how cargo should flow in the channel cargo system (i.e., determine which cargo and how much cargo is on an aircraft during each leg of its mission) in order to minimize the cargo's delay enroute from its origin to its destination. This research showed that since the channel cargo system has a large number of commodities and missions associated with it, the size of an M^2MCF model of the system exceeds AMC's computational capabilities. This research describes three approaches to reduce the problem size. Because of the problem size and other modeling limitations discovered during this research, the presented M^2MCF model of the channel cargo system is currently not accurate enough to be useful as a scheduling tool. However, the M^2MCF model may be adequate for AMC advance planning purposes. Furthermore, the M^2MCF model dual variables may yield useful information for the improvement of AMC's monthly flight schedule. Finally, this research recommends ways to reduce the limitations associated with the M^2MCF model.

DETERMINING CARGO FLOW FOR AIR MOBILITY COMMAND'S CHANNEL CARGO SYSTEM

I. Introduction

I.1 General Issue

There exists a myriad of systems for collecting and delivering goods and services. These systems may involve transporting passengers on a bus, train or other mode of transportation, distributing products between factories and outlets, or collecting and disposing of refuse. A key concern which connects all of these systems is how to efficiently schedule and route available resources to meet customer demands.

There are several ways to measure schedule efficiency with the measure of efficiency selected depending on the objective of the particular problem to be solved. As Bodin observed:

> Usually the objective function is to minimize a weighted combination of capital and operating costs for the fleet [i.e., vehicles used for distribution]. It may also include a formula that represents penalties for not meeting all the time-window constraints and/or for violating other constraints. Also, vehicle routing and scheduling problems can have multiple objective criteria. Sometimes these objectives are hierarchical; in other cases, they are considered concurrently. (Bodin, 1990:574-575)

Likewise, there are several technological constraints which may or may not be considered in a particular problem depending on the assumptions made. These constraints can include: the number of vehicles, vehicle capacity, demand levels for goods and services, and time-window restrictions.

The channel cargo system of the United States Air Force's Air Mobility Command (AMC) is a distribution system in which scheduling and routing must be planned for on a monthly basis. And, as with any other real world problem, the objective function and constraints reflect the required decision making information.

I.2 Background

One of AMC's responsibilities is managing regularly scheduled air service known as the channel network. A channel is a pair of airbases between which AMC must fly to satisfy a military requirement. An AMC channel consists of an origin base and a destination base, known as an origin-destination (OD) pair. The route from the origin base to the destination base may be direct or could have one or more intermediate stops. The channels can be classified into two types: frequency channels and cargo channels. These channel types correspond to the two major types of military requirements that AMC must satisfy: frequency requirements and cargo requirements. A frequency channel is used to provide a minimum number of flights per month between OD pairs. An example is periodic visits to an embassy. A cargo channel is used to transport cargo between OD pairs. The channel cargo system is made up of these two types of channels.

All cargo which cannot be transported using AMC assets must be contracted out to civilian commercial transportation. Since the tonnage of cargo required for shipment varies over time, the Tanker Airlift Control Center (TACC) at AMC must develop its schedules on a monthly basis. These schedules contain the routes and number of missions to be used for that month. This is no small task since in any single month there may be approximately 600 channels based on cargo and 300 channels based on frequency of visit (Ackley et al., 1991:2).

AMC uses a two phase process in their advance planning to determine the number and type of missions needed to be flown for the channel cargo system.

In the first phase of this process, AMC uses a linear programming (LP) model, STORM (Strategic Transport Optimal Routing Model), to determine the number of missions (i.e., routes to be flown by each type of aircraft). STORM's basic purpose is "to select the mix of routes and aircraft that will meet the monthly cargo and frequency requirements while minimizing the costs of cargo handling, military aircraft operations, and commercial aircraft leasing" (Ackley et al., undated:2). Since the solution to the LP model is usually non-integer, AMC uses a heuristic to derive an integer set of missions.

In the second phase, AMC uses a simulation model, CARGOSIM, to validate the results from STORM. Analysis of the CARGOSIM results leads to a schedule that seeks to balance the "dual goals of efficient use of planes and timeliness of delivery." Therefore, "CARGOSIM is used as the sanity check on the linear programming model recommendations regarding a set of missions" (Carter and Litko, undated:1-2). CARGOSIM requires a monthly flight schedule as input. Since STORM only determines the number of missions, AMC uses a simple FORTRAN program called CARGPREP to determine a flight schedule for the routes selected by STORM. CARGPREP divides the number of missions determined by STORM evenly throughout the month (Litko, 9 September 1992). For example, if a mission is to be flown three times that month, then CARGPREP will schedule a mission every 10 days. This schedule along with other sets of known data (i.e., a list of all airbases, a list of all routes to be used for the month, flight times between OD pairs, amount of time required at each stop, and aircraft cargo capacities by aircraft type) is input into CARGOSIM (Hanson, 9 September 1992).

CARGOSIM is written in SIMSCRIPT II.5 as a discrete event model. This model simulates aircraft and cargo flow. The flow of planes is controlled by the input routes and schedules. The generation of cargo is regulated by channel and is modeled as a time dependent Poisson process reflecting the fact that cargo is not generated uniformly throughout the week. The output from CARGOSIM describes channel performance by displaying the mean and variance of the waiting times and travel times for cargo for each OD pair (Carter and Litko, undated:2-3). Timeliness of delivery, expressed in "average delay per cargo ton shipped between each O-D pair" is one of CARGOSIM's primary performance measures (Moul, 1992: 1-5).

An AMC analyst uses the CARGOSIM output to modify the initial schedule produced by CARGPREP. The schedule is modified by changing the flight schedule or increasing the number of missions (Litko, 9 September 1992). The analyst then evaluates the modified schedule using CARGOSIM to determine the amount of cargo which can be delivered on time based on the Uniform Material Movement Issue Priority System (UMMIPS). UMMIPS is a standard used by AMC which dictates the maximum allowable time (in days) a piece of cargo should be in the channel cargo system (Litko, 26 August 1992). This process of schedule modifications and CARGOSIM runs is repeated until the UMMIPS standards are satisfied (Litko, 9

September 1992). This iterative process can take three or four days to complete (Litko, 26 August 1992).

AMC not only uses this two phase process for its advance planning but also uses it for special studies. An example of one such study is analyzing the aerial port structure to determine how changing the number of aerial ports of embarkation and aerial ports of debarkation will impact the routes and missions (Litko, 26 August 1992). AMC could also use this same two phase process to assist the TACC in developing the actual flight schedules.

I.3 Improving the Scheduling Process

Improving a schedule could save AMC money by allowing more cargo to be shipped on time by AMC assets and transporting less by commercial means. This could result in substantial savings since the cost of augmenting AMC aircraft with commercial transport is high -- $148 million was spent in fiscal year 1989 and $165 million was spent in fiscal year 1988 for commercial augmentation (Ackley et al., 1991:2)

In addition, there are some problems associated with AMC's two phase process. Since STORM does not explicitly model timeliness of cargo delivery, "it may shortchange customer service to reduce costs" (Carter and Litko, undated:2). Also, the current process is time-consuming because it takes one analyst at AMC three or four days to develop a schedule using the current, iterative method. Because of the problems associated with the current scheduling process, AMC would like a method which streamlines and improves the process.

This research concentrates on the objective of minimizing the delay enroute. There are two types of delay enroute. The first type is the delay encountered when cargo waits for transportation at the origin base. The second type is the delay which occurs after cargo has left the origin base and includes the flight time and the time that cargo waits for transportation at a transshipment point.

One proposed method to minimize the delay enroute is a two-step, iterative process (Borsi, 6 August 1992). In Step One, given any aircraft schedule, a flow of cargo is determined based on this schedule. The cargo is categorized by its quantity (weight) and its type (origin and destination). Step One will determine the quantity and type of cargo that is loaded or taken off an aircraft as it proceeds from one airbase to another on its assigned route. In Step Two, the aircraft departure times are modified and the schedule revised based on this cargo flow. Returning to Step One with the revised schedule, the cargo flow is modified based on the revised schedule. At each iteration, the delay enroute is reduced. The reduction of the delay enroute after each iteration is used to determine when to stop this iterative process. The two-step process is repeated until the change in the delay enroute is less than or equal to a predetermined criteria.

An obvious advantage of this process is that it uses the output information from STORM and uses the same input data needed by CARGOSIM. This process could be implemented after a schedule is produced by CARGPREP to improve that schedule. The improved schedule can then be used in CARGOSIM. Therefore, this two-step process is compatible with the current scheduling process used by AMC.

I.4 Problem Statement/Research Objective

The purpose of this research is to develop an algorithm which, given a flight schedule and cargo requirements, determines a flow of cargo between OD pairs which minimizes the delay enroute. Specifically, the algorithm designates which cargo and how much cargo is on an aircraft during each leg of its mission. The focus of this research is to minimize the two types of delay enroute. Ultimately, the results of this research can be implemented in the proposed two-step, iterative process described in the previous section to create a better schedule for input to CARGOSIM.

I.5 Assumptions

This section describes the assumptions made in this research. First, all the cargo requirements between OD pairs is known. Additionally, the cargo is classified by weight only and can be divided into an infinite number of subsets. Any other characteristics such as size and urgency of need are assumed to be the same for all cargo (i.e., no outsize cargo and no priority cargo considerations). Passenger requirements will not be considered, and therefore, will not affect the amount of cargo which can be loaded.

The number and type of aircraft available are known and will remain constant (i.e., no breakdowns). Furthermore, each aircraft type will have a specific limitation on cargo weight capacity. Cargo going to different destinations may be loaded on the same aircraft in any proportion as long as the total weight loaded does not exceed the aircraft capacity. Any mixture of cargo is allowed on a single aircraft (i.e., no cargo is considered hazardous). Any cargo can be loaded on any aircraft (i.e., there are no restrictions for a specific cargo to be loaded on a specific aircraft).

Airbases are assumed to be capable of handling an unlimited supply of cargo (i.e., no restrictions on loading machines or storage areas).

Maximizing the cargo load of each aircraft is of secondary importance to minimizing the delay enroute and will not be considered.

I.6 Definitions

Terms used in this research include:

commodity - cargo or OD pair.

mission - a specific type of aircraft flying a specific route.

mission leg - a nonstop path traveled between two airbases.

route - the path traveled by an aircraft from its departure until its return to the homebase.

sortie - one instance of an aircraft flying a specific route which starts and ends at the same airbase. Therefore, a mission flown twice a month represents two sorties for that month.

II. Literature Review

II.1 Scope and Organization of the Review

During an extensive search of journal articles published between 1971 and 1987, Zanakis et al. discovered 127 heuristic methods involving scheduling (Zanakis et al., 1989:88). The purpose of this review is to briefly describe a few of these methods and to present another method, or more specifically, a mathematical model, which AMC can use to create better flight schedules for input to CARGOSIM. The model is the multicommodity network flow model. This review will describe the multicommodity network flow model with emphasis on the multicommodity minimal cost flow model. Additionally, this review will provide examples of how this model has been used to solve some routing and scheduling problems. Finally, this review will describe the dual variable, which may provide information to improve AMC's monthly flight schedule.

II.2 Methods to Create Better Schedules

Several methods have been developed which would help AMC create better schedules. These methods reduce or eliminate schedule inefficiencies such as excessive cost (Gertsbakh and Serafini, 1991:298), excessive delay (Solanki and Southworth, 1991:124), and insufficient use of the transporting vehicle (Kikuchi and Rhee, 1989:643). As stated in Chapter I, measuring schedule efficiency depends on the objectives of the organization. Likewise, these methods are tailored around the objective. For example, Gertsbakh and Serafinis' objective for schedule construction is to minimize the cost of shipping the goods from the origin to the destination by minimizing the fleet size needed to transport the cargo (Gertsbakh and Serafini, 1991:298). Kikuchi and Rhees' objective is to maximize vehicle use by maximizing the number of trips assigned to each vehicle (Kikuchi and Rhee, 1989:643). Still another objective, and the one which this research uses, is to minimize the delay enroute.

II.3 Multicommodity Network Flow Problems

Multicommodity network flow problems (MNFP) are specially structured linear programming problems which "arise when several items (commodities) share arcs in a capacitated network" (Kennington, 1978:209). The problem can be described on a network made up of nodes and arcs. Each commodity is identified by its source (origin) and its sink (destination) (Wollmer, 1972:247). The advantage of formulating a problem as a MNFP as opposed to a general linear program is that specialized multicommodity network flow computer programs can solve the problem faster than a general LP solver (Ali et al., 1984:127). Two types of MNFP are the multicommodity minimal cost flow (MMCF) problem and the multicommodity maximum flow (MMF) problem. Kennington describes the MMCF problem as:

> [a problem whose objective is] to determine a minimal cost multicommodity flow through a network that meets the demand for each commodity, subject to (i) supply restrictions, (ii) arc capacity restrictions, and (iii) flow conservation at transshipment nodes. (Kennington, 1978:210)

The MMF problem's objective is to find the maximum, nonnegative flow of all commodities in the network subject to (i) arc capacity restrictions and (ii) flow conservation at transshipment nodes (Kennington, 1978:210).

The MMCF has been used to solve many routing and scheduling problems. White and Wrathall applied the MMCF model to solve the problem of scheduling railroad cars between their origin and destination points (White and Wrathall, 1970:1). Their objective was "to minimize the total elapsed time for the cars requiring movement on the railroad...subject to the capacity of the yards and the trains themselves" (White and Wrathall, 1970:17).

Bellmore, Bennington and Lubore used a variation of the MMCF to solve a multivehicle tanker scheduling problem. The objective was to maximize the utility of a fixed fleet of tankers in making a specified set of shipments (Bellmore et al., 1971:37).

Clarke and Surkis solved a racial desegregation problem for school systems using the MMCF model. Their objective was to minimize student transportation time subject to achieving a desired ethnic composition at each school and ensuring that no student traveled more than a specified amount of time per day (Clarke and Surkis, 1968:259).

II.4 The Multicommodity Minimal Cost Flow Problem

As shown in the previous section, the MMCF model has been used to solve a variety of routing and scheduling problems. This section will provide a more detailed discussion of the MMCF model.

Kennington describes a multicommodity network as a network [V,E] consisting of "a finite set V of nodes $1,\ldots,N$, and a finite set E of ordered pairs of nodes, $e_m=(i,j)$, called arcs" (Kennington, 1978:209). Furthermore, there are K commodities with each commodity k having a single source s_k and a single sink t_k with a supply and demand of S_k, $k=1,\ldots,K$. Kennington expresses the mathematical formulation of the MMCF as follows (Kennington, 1978:210):

The objective function for the MMCF model is:

$$Min \sum_k \sum_m c_m^k x_m^k \qquad (1)$$

where c_m^k and x_m^k are the unit cost and flow, respectively, for commodity k in arc e_m.

The constraints for conservation of flow are expressed mathematically for each node n as:

$$\sum_{e_m \in A_n} x_m^k - \sum_{e_m \in B_n} x_m^k = \begin{cases} +S_k, & if\ n=s_k \\ -S_k, & if\ n=t_k \\ 0, & otherwise \end{cases} \tag{2}$$

where A_n is the set of arcs that originate at node n, and B_n is the set of arcs that terminate at node n.

The constraints which limit the sum of the flows of all commodities on each arc m are expressed as:

$$\sum_k x_m^k \le b_m \tag{3}$$

where b_m is the capacity of arc e_m. Ali et al. noted that these "constraints link the commodities and are called linking constraints or generalized upper bounding (GUB) constraints" (Ali et al., 1984:128).

Finally, the constraints which limit the flow of each commodity k on arcs are expressed as:

$$0 \leq x_m^k \leq u_m^k \tag{4}$$

where u_m^k is the maximum amount of commodity k which can flow on arc e_m.

Helgason and Kennington note that the constraint matrix of an MMCF model "assumes the block angular form" (Helgason and Kennington, 1977:298). An example of the constraint matrix for an MMCF model is shown in Figure 1. The constraint matrix A of the MMCF model can be divided into two groups: the $A(\cdot)$ matrix, and the *GUB* coupling constraints. The $A(\cdot)$ matrix consists of K node-arc incidence matrices A_k. In other words, each A_k matrix is replicated K times -- one node-arc incidence matrix A_k for each commodity k. Each matrix A_k represents a subgraph of the network. The *GUB* constraints consist of a row of K identity matrices I.

Helgason and Kennington explain that the MMCF model "can be generalized to allow for commodity-dependent subgraphs (instead of using [V,E] for each commodity)" (Helgason and Kennington, 1977:298). They further explain that such a generalization "involves no mathematical difficulties, but greatly complicates the notation" (Helgason and Kennington, 1977:298). Therefore, each

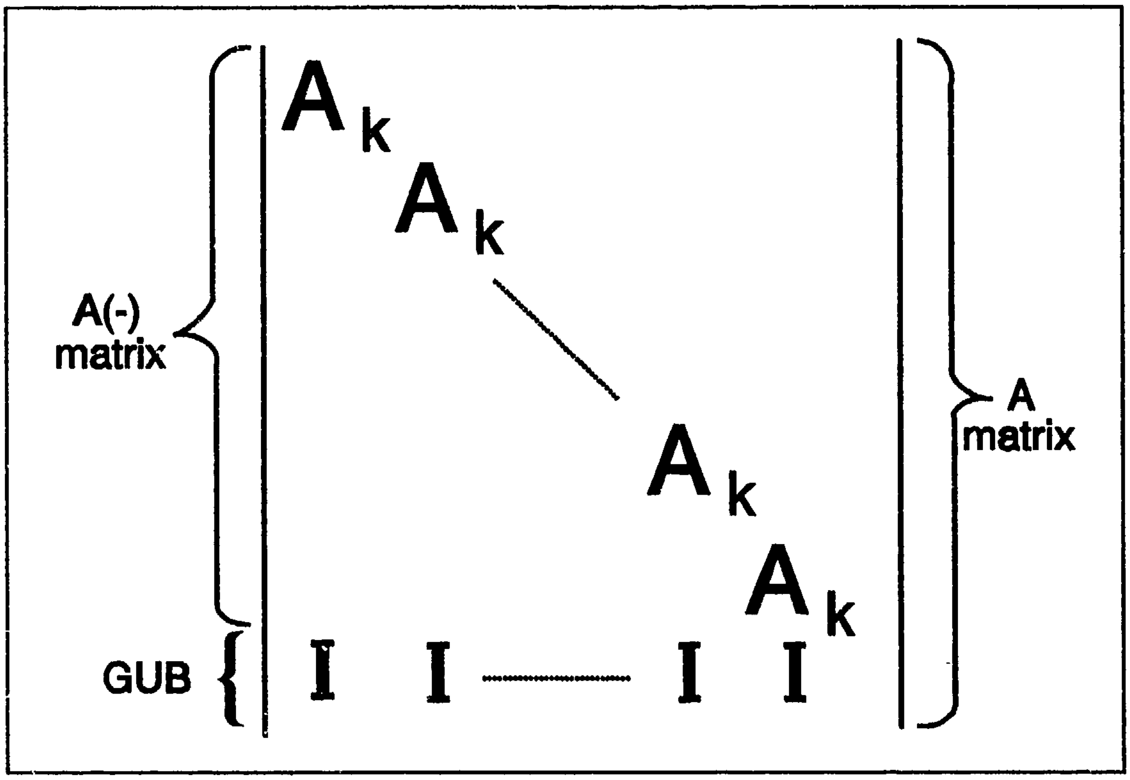


Figure 1

matrix A_k of the constraint matrix can be "tailored" to the particular commodity with which it is associated, i.e., not every node and arc in the original network [V,E] needs to be replicated in any given subgraph.

II.5 The Dual Variable

"Associated with any LP [linear program] is another LP, called the dual." Furthermore, when "taking the dual of a given LP, we refer to the given LP as the primal". The value of the dual variable w_i is commonly called the marginal cost or the shadow price of the ith primal constraint. The shadow price of the *i*th constraint is the rate at which the optimal objective function value can be improved (increased in a maximization problem and decreased in a minimization problem) if the value of the right-hand-side of the *i*th constraint in the primal LP is increased by a small amount. Additionally, the dual variables only provide reliable information over a specific range and when dealing with a change in the right-hand-side value of a single constraint. Furthermore, the dual variable is difficult to interpret when degeneracy exists (Bazaara et al., 1990:256-259; Borsi, 8 February 93; Winston, 1991:271,272,292).

II.6 Conclusion

The MNFP model, and in particular the MMCF model, is one model which can be used to solve particular routing and scheduling problems. Several examples were presented earlier to show this. Chapter III describes how the MMCF formulation is used to model the channel cargo system. Additionally, the dual variable of a linear program provides information on the rate of change of the objective function value for small changes in the right-hand-side value of a primal constraint. Chapter IV describes how the dual variables may provide information as to how to modify the flight schedule of the channel cargo system to improve the objective function value.

III. Methodology

III.1 General

The AMC channel cargo system can be viewed as a network problem. A network problem is a problem that can be represented by a set of nodes and a set of edges or arcs which connect the nodes. The arcs may have direction and flows associated with them. Technological constraints may be included to restrict the amount of flow through the arcs. For example, if the channel cargo system is viewed as a network, each node represents an airbase while each arc represents a mission leg. For this research, the channel cargo system has been modeled as a multicommodity network flow problem. As explained in Chapter I, the purpose of this research is to determine a flow of cargo for Step One of the proposed schedule improvement process which minimizes the delay enroute. Modeling the channel cargo system using a multicommodity network will allow one to determine a flow of cargo which minimizes total transit time for all commodities.

III.2 The Multiperiod Characteristic of the Channel Cargo System

The multicommodity network flow models discussed in Chapter II were models that can represent systems for a single period of time. One way to illustrate this is by defining the flow variable, x_m^k, and the unit cost variable, c_m^k, described in Chapter II, in terms of what they represent in the AMC channel cargo system. Recall that x_m^k is the amount of commodity k flowing in arc e_m, and c_m^k is the unit cost for commodity k in arc e_m. In modeling the channel cargo system, each node could represent an airbase, and each arc could represent an aircraft traveling from one airbase to another. Therefore, there would be one node for each airbase in the system and one arc for each mission leg. The flow variable x_m^k would represent the amount of cargo of a specific OD pair which was being transported between airbases on a particular aircraft, while the unit cost variable c_m^k would represent the time required to transport this cargo on the aircraft from one airbase to another. However, the limitation of this type of formulation is that cargo which must remain at an airbase to await transport and the associated delay caused by this wait is not modeled. This limitation applies to both origin bases and transshipment bases. AMC is interested in delay enroute caused by both the time associated with transporting cargo on an aircraft and the delay associated with cargo awaiting transportation at an airbase. One way to account for both types of delay enroute is to create a multicommodity multiperiod network (Borsi, 28 August 1992).

III.3 Example of a Multicommodity Multiperiod Network

This section will provide a few illustrations to explain how the channel cargo system can be modeled using a multicommodity multiperiod (MM) network. Consider a two airbase system with one aircraft transporting cargo between airbase *a* and airbase *b*. The route for this aircraft is *a*-*b*-*a* (start at *a*, fly to *b*, and return to *a*). The aircraft can fly this route in six hours; therefore, let the planning horizon under consideration be six hours. Two equivalent network representations of this system are shown in Figures 2(a) and 2(b). The nodes represent the two airbases, and the arcs represent the aircraft flying between the two airbases. Both of these networks only consider the delay associated with transporting cargo and not the delay associated with cargo awaiting transportation.

This same two airbase system can also be represented using an network as shown in Figure 3(a). Note that in the MM network the planning horizon is divided into two time increments of three hours each. Additionally, each airbase is represented at three time periods (u=1,2,3) to represent the airbases at the beginning of the planning horizon, at each consecutive time increment, and at the end of the planning horizon. For example, airbase *a* is represented three times (by nodes a1, a2, and a3) to correspond to the three separate time periods (u=1,2,3), respectively. The

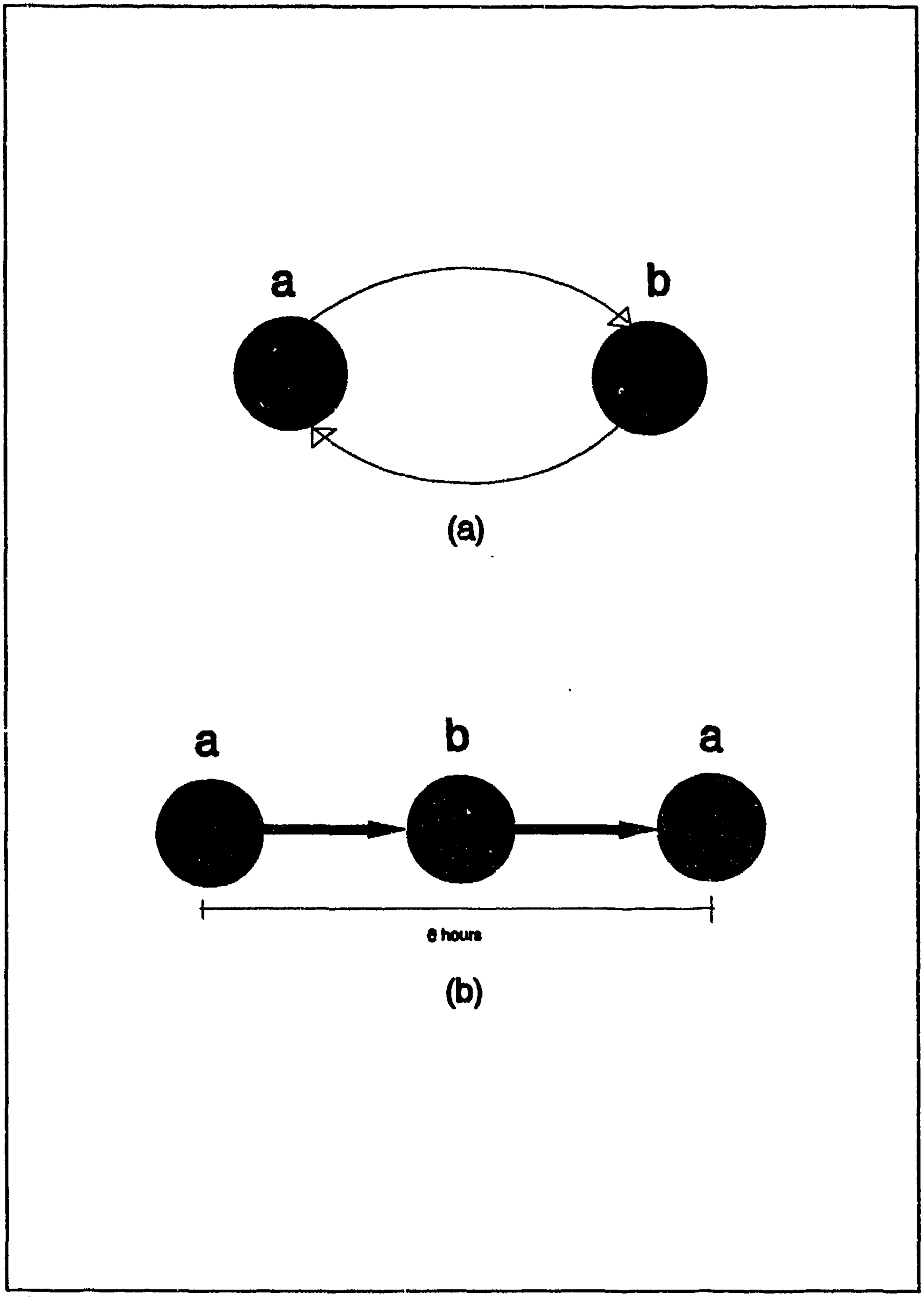
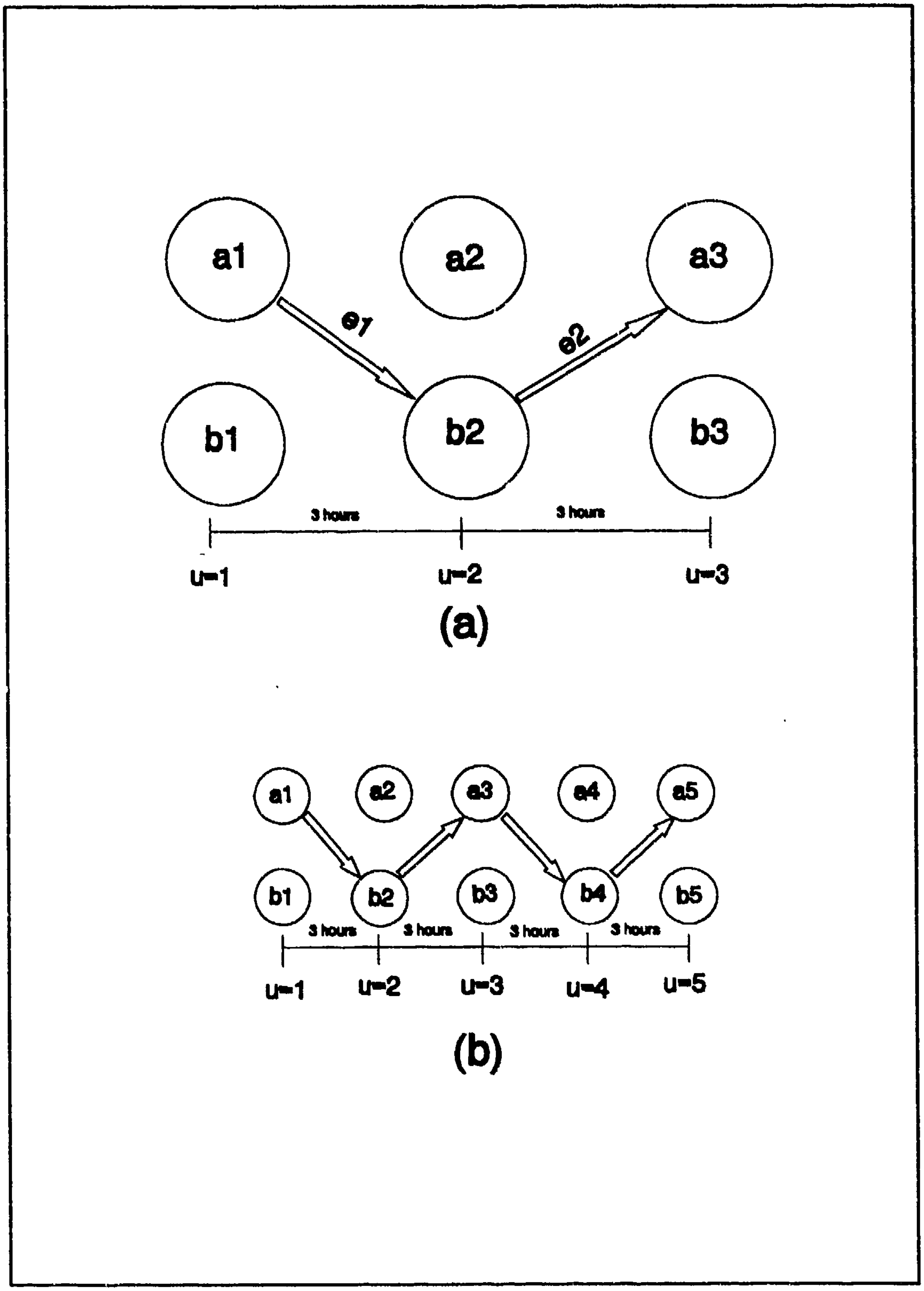


Figure 2

arc e_1 represents the aircraft departing airbase *a* and arriving at airbase *b* between time periods u=1 and u=2. Likewise, the arc e_2 represents the same aircraft departing airbase *b* and arriving at airbase *a* between time periods u=2 and u=3. Although they are not shown in the figure, arcs between airbases in the same time period (such as an arc from a1 to b1) and arcs between airbases which connect time periods that are not consecutive (such as an arc from a1 to b3) are permissible if an aircraft makes the indicated trip within that range of time periods.

If a longer planning horizon (i.e., twelve hours) is desired, then the corresponding MM network would look like the one shown in Figure 3(b). In Figure 3(b), there is still only one mission; however, the frequency of this mission (i.e., the number of times a mission is flown during the planning horizon) has doubled. Therefore, this figure shows two sorties flown during the planning horizon.

Figures 3(a) and 3(b) show only a portion of a complete MM network. Consider two commodities for this system: the cargo required to be transported from airbase *a* to *b* (commodity *ab*), and the cargo required to be transported from airbase *b* to *a* (commodity *ba*). Because of capacity limitations on the aircraft, not all of the cargo may fit on the aircraft at any one time. Therefore, some of the cargo must remain at the airbase until an aircraft is available to



a1
a2
a3
e1
e2
b1
b2
b3
3 hours
3 hours
u=1
u=2
u=3
(a)
a1
a2
a3
a4
a5
b1
b2
b3
b4
b5
3 hours
3 hours
3 hours
3 hours
u=1
u=2
u=3
u=4
u=5
(b)

Figure 3

carry more cargo. This situation can be modeled as a set of parallel, horizontal arcs connecting the same airbases over time as shown in Figure 4(a). Flow on arc e_3 represents cargo at airbase *a* at time period 1 which must remain at airbase *a* until time period 2. The other arcs are interpreted in a similar manner.

The complete MM network for the two airbase system is shown in Figure 4(b). Note that the networks shown in Figures 2(a) and 2(b) do not represent cargo which must remain at a particular airbase to await transportation, and therefore, fail to model the two types of delay enroute which are of interest to AMC. The MM network, however, can model both types of delay enroute in a single network.

III.4 Steady State Conditions in an MM Network

If the mission for this two airbase system is repetitive, and the aircraft flies the route *a-b-a* every six hours, then the system can be modeled by replacing the a3 and b3 nodes (shown in Figure 4(b)) by a1 and b1, respectively, as shown in Figure 5(a).

This **steady state** representation reflects what the analysts at AMC do when they use CARGOSIM. When using CARGOSIM, the analysts must replicate the monthly flight schedule three times. Of the three monthly schedules, the second and the third schedules are the ones that are

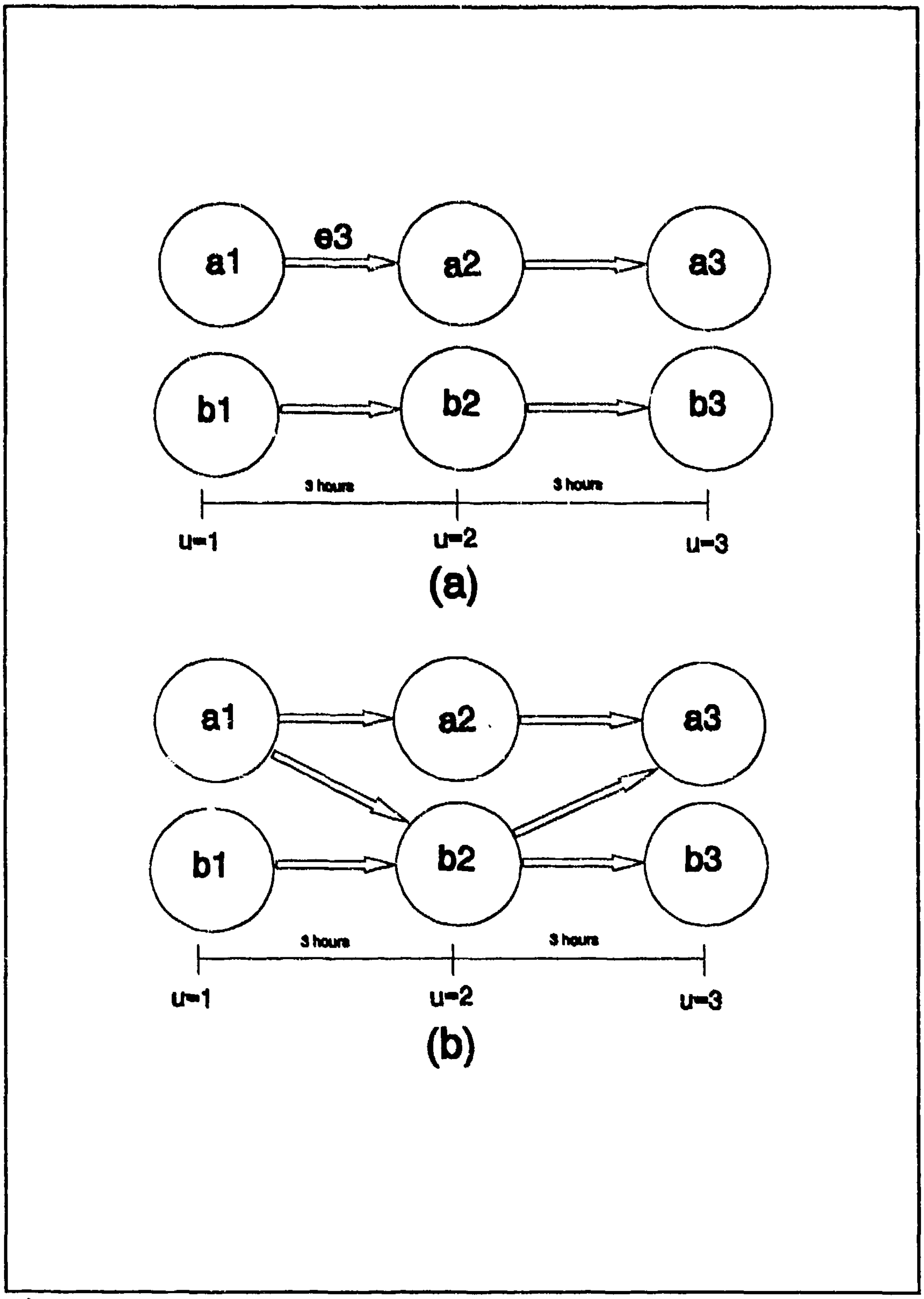

a1
e3
a2
a3
b1
b2
b3
3 hours
3 hours
u=1
u=2
u=3
(a)
a1
a2
a3
b1
b2
b3
3 hours
3 hours
u=1
u=2
u=3
(b)

Figure 4

studied. The purpose of the first monthly flight schedule is to generate cargo and simulate the backlog of cargo which is awaiting transportation prior to the start of the months under study (Robinson, 22 Sep 92). The steady state representation performs the same function by returning undelivered cargo to the beginning of the time horizon.

III.5 Commodity Arrival Times

Cargo in the channel cargo system does not arrive uniformly throughout the week. On the average, cargo arrival is light at the beginning of the week and peaks slightly after mid-week (Carter and Litko, undated:2). However, the cargo generation is assumed to be the same from one week in a given month to the next week in the same month (Whisman, 22 September 1992).

The expected values of arriving commodities can be shown on the MM network by displaying the amount of the commodity arriving at a given airbase at a particular time period in brackets above the appropriate node as shown in Figure 5(b). Figure 5(b) shows that three units of commodity *ab* arrive at time period 1 and are available for transport at time period 1. The figure also shows that four units of commodity *ab* arrive at time period 2 and are available for transport at time period 2.

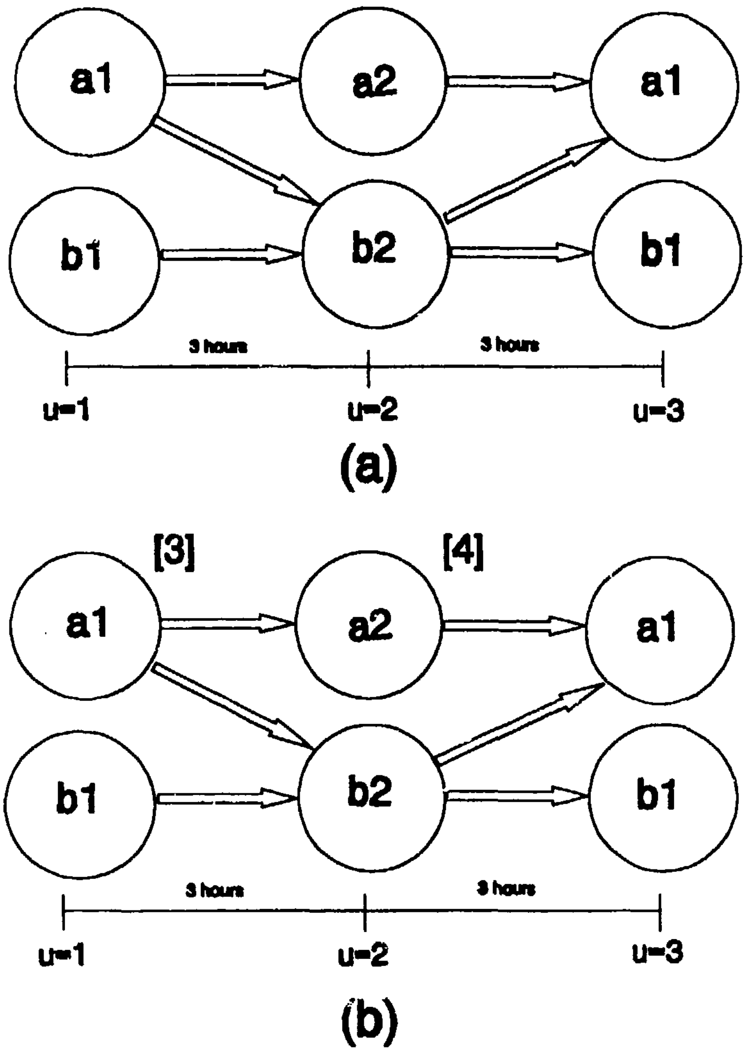

a1
a2
a1
b1
b2
b1
3 hours
3 hours
u=1
u=2
u=3
(a)
[3]
[4]
a1
a2
a1
b1
b2
b1
3 hours
3 hours
u=1
u=2
u=3
(b)

Figure 5

III.6 The Channel Cargo System Modeled as a Multiperiod Multicommodity Minimal Cost Flow Problem

Based on the examples discussed in the previous sections, one can now describe and formulate the channel cargo system in terms of a multiperiod multicommodity minimal cost flow (M^2MCF) problem. The notation needed for the problem description and formulation is as follows:

<u>Indices</u>

a = arc index.

d = airbase index signifying destination base.

i, j, k = airbase indices.

o = airbase index signifying origin base.

u, v, w = time period indices.

<u>Sets</u>

A_{iu} = set of arcs that originate at node n_{iu}.

B_{iu} = set of arcs that terminate at node n_{iu}.

E = finite set of all arcs.

ES = subset of set E representing mission legs.

ET = subset of set E consisting of the arcs connecting the same airbases from one period to the next period.

K = finite set of all commodities k_{od}.

T = finite set of all the time period indices u.

V = finite set of nodes which represent airbases at particular periods in time.

Network Data

$b_{(iu,jv)}$ = the capacity of the aircraft traveling between airbases i and j between time periods u and v, $i \neq j$.

$c_{(iu,jv)}{}^{od}$ = unit cost of transporting commodity k_{od} from node n_{iu} to node n_{jv}.

DM_{od} = the maximum demand of commodity k_{od} at airbase d for any given time period.

e_a = arc a in set E, $e_a = (n_{iu}, n_{jv})$, $u \leq v$.

$|K|$ = total number of commodities/cargo types.

k_{od} = commodity which must be transported from origin base o to destination base d.

n_{iu} = node representing airbase i at time period u.

si_{uod} = node n_{du} which serves as a sink node for commodity k_{od} at time period u.

so_{uod} = node n_{ou} which serves as a source node for commodity k_{od} at time period u.

sp_{uod} = the amount of commodity k_{od} which is initially ready for shipment at airbase o at time period u.

Variable

$x_{(iu,jv)}{}^{od}$ = amount of commodity k_{od} in transit from node n_{iu} to node n_{jv}.

The channel cargo system, therefore, can be expressed as a finite set V of nodes and a finite set E of arcs. The set E can be partitioned into two mutually exclusive, totally exhaustive subsets ES and ET. ES is the set of arcs representing mission legs. ET is the set of arcs connecting the same airbase from one time period to the next. The flow on an arc $e_a \in ET$ represents the commodities which remain at airbase i from time period u to time period v (awaiting transportation).

The channel cargo system has $|K|$ commodities, each designated by k_{od}. Any particular commodity k_{od} has multiple sources so_{uod} and multiple sinks si_{uod}. sp_{uod} is the amount of commodity k_{od} ready for shipment at time period u. For example, in Figure 5(b), sp_{1ab}=3 and sp_{2ab}=4. DM_{od} is the maximum demand for commodity k_{od} in any given time period.

The mathematical formulation of the M^2MCF problem is described below. The objective function is:

$$Min \sum_{k_{od} \in K} \sum_{(n_{iu}, n_{jv}) \in E} C^{od}_{(iu,jv)} X^{od}_{(iu,jv)} \tag{5}$$

The constraints are:

$$\sum_{(n_{iu},n_{jv})\in A_{iu}} x_{(iu,jv)}^{od} - \sum_{(n_{kw},n_{iu})\in B_{iu}} x_{(kw,iu)}^{od} = \begin{cases} sp_{uod}, \ \forall\, n_{iu}=so_{uod},\ k_{od}\in K \\ 0,\ \forall\, n_{iu}\neq so_{uod} \text{ or } si_{uod}, \\ \quad k_{od}\in K \end{cases} \quad (6)$$

$$\sum_{(n_{iu},n_{jv})\in A_{iu}} x_{(iu,jv)}^{od} - \sum_{(n_{kw},n_{iu})\in B_{iu}} x_{(kw,iu)}^{od} \geq -DM_{od},\ \forall\, n_{iu}=si_{uod},\ k_{od}\in K \quad (7)$$

$$\sum_{k_{od}\in K_{od}} x_{(iu,jv)}^{od} \leq b_{(iu,jv)},\ \forall\, e_a\in ES \quad (8)$$

$$x_{(iu,jv)}^{od} \geq 0,\ \forall\, e_a\in E,\ k_{od}\in K \quad (9)$$

The unit cost, $c_{(iu,jv)}{}^{od}$, in Equation (5) is the transit time required for commodity k_{od} to go from node n_{iu} to node n_{jv}. For arcs $e_a \in ES$, the unit cost is the flight time for that particular mission leg. For arcs $e_a \in ET$, the unit cost is the time increment between time periods.

Equation (6) is the set of conservation of flow constraints. The expression equals sp_{uod} if the node n_{iu} is a source node so_{uod}, and the expression equals zero if the node n_{iu} is neither a source node so_{uod} nor a sink node si_{uod}.

Equation (7) is the set of modified conservation of flow constraints for sink nodes si_{uod}. Since the actual demand (i.e., the actual amount of a commodity delivered) at an airbase at a particular period in time is not known, Equation (7) is an inequality. The flow at sink nodes is less than or equal to DM_{od} where DM_{od} is calculated in Equation (10) below. Equation (7), therefore, allows a sink node to demand the optimal number of flow units.

Equation (8) is the set of General Upper Bounding (GUB) constraints which limit the sum of the flows of all commodities on a given aircraft. The constraint is necessary only for arcs $e_a \in ES$. The assumption that airbases are capable of handling an unlimited supply of cargo eliminates the need to have additional constraint equations which consider arcs that are contained in subset ET.

Equation (9) is the set of nonnegativity constraints.

Note that there is no set of constraints similar to Equation (4) in Chapter II. These upper bound constraints on the flow along an arc are not necessary since one of the assumptions made in Chapter I was that cargo going to different destinations may be loaded on the same aircraft in any proportion as long as the total weight loaded does not exceed the aircraft capacity. Therefore, the only upper bound limit on the flow of commodities is the capacity of the aircraft, and this is modeled in Equation (8).

As mentioned earlier, Equation (10) is used to calculate the maximum demand DM_{od} at an airbase which serves as a sink node:

$$DM_{od} = \sum_{u \in T} sp_{uod}, \ \forall \ k_{od} \tag{10}$$

The maximum demand DM_{od} is the sum of all the supplies of a commodity for the different periods in the planning horizon and is an upper bound on the demand for each commodity.

III.7 Problem Size of an M^2MCF Problem

The size of an M^2MCF problem can be defined as the number of variables and the number of constraints needed to model the problem using the M^2MCF formulation. The size of an M^2MCF problem can be determined using the following additional notation:

A = the node-arc incidence matrix of the multicommodity network.

AB = the number of airbases in the system.

C_{max} = the maximum number of constraints in the M^2MCF problem.

C_{maxest} = the estimated maximum number of constraints in the M^2MCF problem.

GUB_{tot} = the total number of GUB constraints in the system, i.e., the total number of mission legs flown in the channel cargo system during the planning horizon.

GUB_{est} = the estimated number of GUB constraints in the system.

leg_{avg} = average number of legs in a mission.

leg_a = 1 if there exists an arc $e_a \in ES$.

= 0 otherwise.

N_{max} = the maximum number of nodes in the M^2MCF problem.

srt = the total number of sorties flown during the planning horizon. For example, a mission flown twice during a month represents two sorties for that month.

t_{tot} = the total number of time periods in the planning horizon.

VAR_{max} = the maximum number of variables (i.e., the maximum number of arcs) in the M^2MCF problem.

VAR_{maxest} = the estimated maximum number of variables in the M^2MCF problem.

The maximum number of nodes N_{max} in the MM network can be calculated using the formula below:

$$N_{max} = (AB)(t_{tot}) \tag{11}$$

The maximum number of variables VAR_{max} in the MM network can be determined using the following formula:

$$VAR_{max} = (|K|)(GUB_{tot}) + (|K|)(N_{max}) \tag{12}$$

where N_{max} is calculated according to Equation (11) above and GUB_{tot} is calculated according to Equation (13) below. The $(|K|)(GUB_{tot})$ term in Equation (12) determines the number of possible commodity-arc combinations for arcs $e_a \in ES$. The $(|K|)(N_{max})$ term in Equation (12) determines the number of possible commodity-arc combinations for arcs $e_a \in ET$ (assuming that a steady state system, as described in Section III.4, is modeled). Using a steady state system, the number of arcs $e_a \in ET$ for any one base is equal to the number of time periods t_{tot}. Therefore, the number of arcs $e_a \in ET$ for all airbases is equal to $(AB)(t_{tot})$ or N_{max} (using Equation (11)).

The total number of GUB constraints in the problem GUB_{tot} can be calculated as follows:

$$GUB_{tot}=\sum_{a} leg_{a}, \text{ for } e_{a} \in ES \tag{13}$$

GUB_{tot} is dependent on the output from STORM and this varies from month to month depending upon the cargo generation. Therefore, a way to estimate GUB_{tot} is given below:

$$GUB_{est}=(srt)(leg_{avg}) \tag{14}$$

Equation (12) can now be rewritten as:

$$VAR_{maxest}=(|K|)(GUB_{est})+(|K|)(N_{max}) \tag{15}$$

The maximum number of constraints C_{max} can be calculated as follows:

$$C_{max} = (N_{max})(|K|) + GUB_{tot} \tag{16}$$

The estimated maximum number of constraints C_{maxest} is obtained by substituting GUB_{est} for GUB_{tot}. Therefore:

$$C_{maxest} = (N_{max})(|K|) + GUB_{est} \tag{17}$$

III.8 Determining Problem Size for the Channel Cargo System

Based on the formulas given in the previous section, the size of the channel cargo system, if modeled as an M^2MCF, can now be estimated. The following data is typical for the channel cargo system for any given month (Whisman, 22 September 1992):

AB = 169 airbases

$|K|$ = 437 commodities (o-d pairs)

srt = 528 (per month)

leg_{avg} = 3 legs per mission

Considering a planning horizon of T=30 days and a time increment of 1 day, there will be t_{tot}=30 time periods (i.e., each day represents a time period). Therefore, the maximum number of nodes can be determined using Equation (11): N_{max}=(169)(30)=5070.

The estimated number of GUB constraints can be found using Equation (14): $GUB_{est}=(528)(3)=1584$.

The estimated maximum number of variables can be found using Equation (15):

$VAR_{maxest}=(437)(1584)+(437)(5070)=2,907,798$.

Finally, the estimated maximum number of constraints can be determined using Equation (17):

$C_{maxest}=(5070)(437)+1584=2,217,174$.

AMC's Force Structure Analysis office is capable of solving a linear programming problem which has 160,000 variables and 20,000 rows (Whisman, 30 October 1992). The number of variables and constraints for the channel cargo system, if modeled as an M^2MCF problem, exceeds this capability. Therefore, this M^2MCF formulation of the entire channel cargo system cannot be solved with AMC's current computer resources.

III.9 Reducing the Problem Size

Three approaches to reduce the number of variables and constraints for an M^2MCF model of the channel cargo system have been examined. These three approaches, when combined, reduce the problem size to one which AMC can solve. The first approach is to break down the channel cargo system into separate geographic areas and solve a M^2MCF problem for each geographic area. The personnel at AMC say that using this approach, the channel cargo system should be divided according to the amount of interaction between the U.S. airbases and the airbases of other geographic areas. This interaction is a function of the number of OD pairs between and within the different geographic areas. Therefore, it appears that the best way to divide the channel cargo system is to have airbases from the U.S. interact with airbases in the following four areas: 1) the Pacific (which includes Australia, New Zealand, Japan, Korea and Indonesia), 2) Europe/Southwest Asia (which includes Iceland and Greenland), 3) Africa (which includes Diego Garcia), and 4) the Americas (which includes Canada, Central America, South America, and the Caribbean and does not include the U.S.) (Litko, 13 October 1992). Therefore, the overall problem can be broken into four smaller M^2MCF problems which represent the OD pairs and missions associated with the U.S. and each of the four geographic areas listed above.

This approach appears reasonable since there is substantially more interaction between the U.S. and the other four areas compared to the interaction between any two of the four areas. For example, in a recent AMC study involving a total of 435 OD pairs, there were 176 OD pairs associated with the U.S./Pacific area, 147 pairs associated with the U.S./Europe/Southwest Asia area, 11 for the U.S./Africa area, and 92 for the U.S./Americas area. The only intra-theater interactions were 1 OD pair between

Europe and Africa and 8 OD pairs between Europe and the Pacific (Whisman, 27 Oct 92). With the exception of the U.S./Africa area, the intra-theater interaction was minimal compared to the U.S./inter-theater interaction.

The second approach to reduce the problem size is based on an observation of Helgason and Kennington presented in Chapter II. To reiterate, they say that the constraint matrix of the multicommodity minimal cost flow (MMCF) model "assumes the block angular form" (Helgason and Kennington, 1977:298). An example of the constraint matrix for an MMCF model is shown in Figure 6.

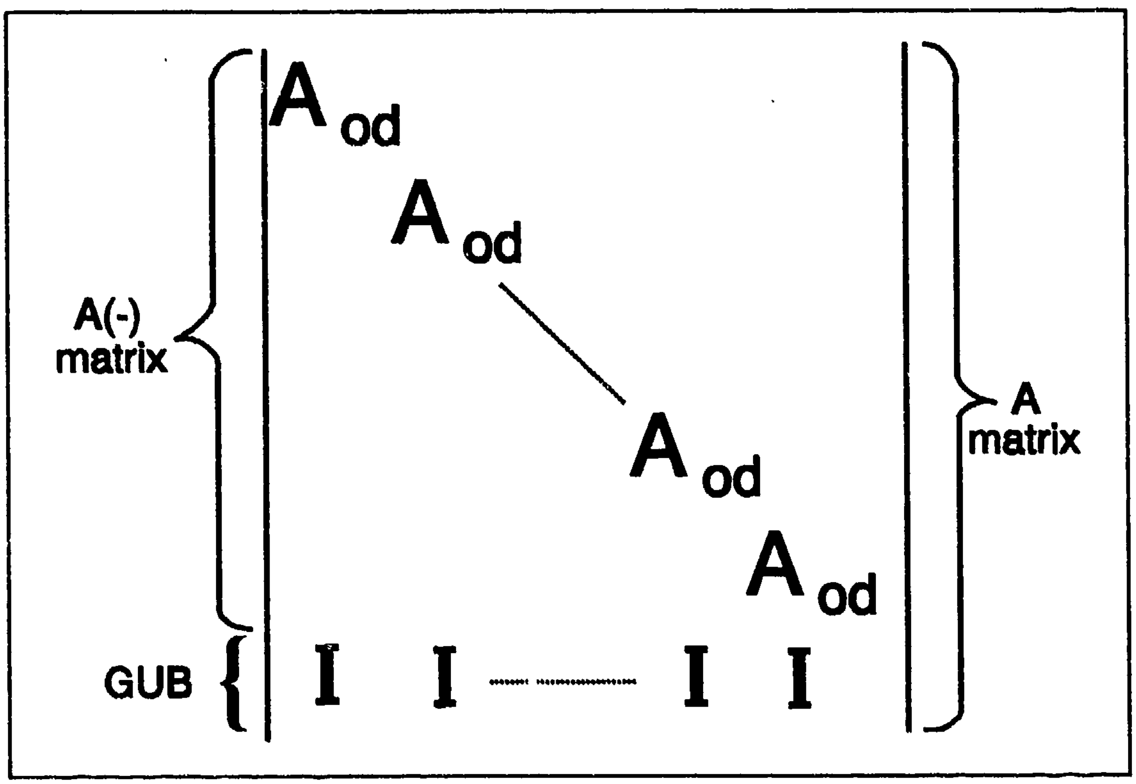


Figure 6

The constraint matrix A of the MMCF model can be divided into two groups: the $A(\cdot)$ matrix and the *GUB* coupling constraints. The $A(\cdot)$ matrix consists of $|K|$ node-arc incidence matrices A_{od}. In other words, each A_{od} matrix is replicated $|K|$ times -- one node-arc incidence matrix A_{od} for each commodity k_{od}. Each matrix A_{od} represents a subgraph of the network. The *GUB* constraints consist of a row of $|K|$ identity matrices I. Helgason and Kennington further say that the MMCF model "can be generalized to allow for commodity-dependent subgraphs..." (Helgason and Kennington, 1977:298). With respect to the modeling of the channel cargo system, this means that all the mission legs for a given geographic area need not be represented in each and every matrix A_{od} of the M^2MCF constraint matrix.

This idea of limiting the arcs, which represent the mission legs, for each subgraph depending on which commodity is being shipped fits well with the assumptions that the AMC personnel make in their STORM and CARGOSIM models. The STORM and CARGOSIM models assume that only certain commodities can be transshipped, these commodities can be transshipped only once, and the possible transshipment points for these commodities are known (Whisman, 22 September 1992). With these assumptions, it is easy to identify which mission legs should be included in the subgraph for any particular commodity.

The third approach for reducing the problem size is to alter the length of the planning horizon and the length of the time increment. Ideally, the planning horizon should be 30 days since this is the length of the schedule that AMC usually studies (Whisman, 22 September 1992). However, since the cargo generation is assumed to be the same from one week in a given month to the next week in the same month (Whisman, 22 September 1992), a seven day planning horizon may be reasonable. The number of time periods in the planning horizon depends on the desired degree of accuracy needed for a valid depiction of the flight schedule. For example, for a seven day planning horizon with one time period per day (i.e., a time increment of 24 hours), any flights arriving and departing a given airbase during that entire day will be represented. However, if the time increment is eight hours, then only the aircraft arriving and departing a given airbase during that eight hour period will be represented. Therefore, a tradeoff must be made between the accurate portrayal of the channel cargo system (by using smaller time increments) and the problem size (which increases as the size of the time increments decrease). Based on discussions with the AMC analysts, a time increment of eight hours should be appropriate (Whisman, 22 September 1992).

III.10 Revised Problem Size for the Channel Cargo System

Based on the three approaches outlined in the previous section, the problem size for the channel cargo system can be revised. For this research, only one of the four geographic areas in the channel cargo system was considered -- the Europe/Southwest Asia area. There are two major reasons for selecting this area to study. First, the Europe/Southwest Asia and the Pacific areas are substantially larger than the Americas and the Africa areas when considering such factors as the number of commodities requiring transport, the number of routes, and the number of mission legs in those areas (Robinson, 22 Sep 92). Second, the likelihood that commodities will be transshipped and the number of occurrences of these transshipments in the Europe/Southwest Asia area are greater than in the Pacific area (Whisman, 22 Sep 92). The calculations for the problem size of the Europe/Southwest Asia (E/SWA) area, using the approaches described in the previous section to reduce the problem size, are shown in Appendices A through D. These appendices show that, given a planning horizon of one week and considering 21 time periods in that week, the number of variables required to model the E/SWA area is 66,395 and the number of constraints required is 11,681. Therefore, the reformulated problem size does not exceed the computer capabilities of the AMC Force Structure Analysis office.

III.11 Modeling a Portion of the E/SWA Area using the M^2MCF Formulation

The E/SWA area consists of 40 airbases, 145 commodities (i.e., OD pairs), 49 routes, and 295 mission l gs (for a one week period) (Robinson, 22 Sep 92). Because the VAX/VMS computer system at the Air Force Institute of Technology is not able to handle a problem with these dimensions, a smaller subproblem was formulated using an extract of the information from the E/SWA area. Only 36 of the 40 airbases, 20 of the 145 commodities, 37 of the 49 routes, and 257 of the 295 mission legs were chosen for this subproblem. This yields a problem with 20,001 variables and 15,422 constraints. All of the commodities chosen required a transshipment.

With the selection of the subproblem, the M^2MCF formulation presented in Section III.6 was written in a computer program using the General Algebraic Modeling System (GAMS) language. The GAMS program is written in such a manner that, given enough computer memory, additional airbases, commodities, and routes can easily be added to the problem. The subproblem input data for the GAMS program is shown in Appendices E through I. This input data is in the same format required for the STORM and CARGOSIM models. Since the input data format was not suitable for the GAMS program, a FORTRAN program was used to pre-process and reformat the data and to write the GAMS program into a file.

This FORTRAN program, "GAMS.FOR", is shown in Appendix J. In addition to creating the GAMS program (shown in Appendix K), the GAMS.FOR program also creates two temporary data files which are shown in Appendices L and M. A partial listing of the results from the GAMS program used to solve the subproblem is shown in Appendix N.

III.12 Analysis of the Results

Since the solution to the M^2MCF formulation of the subproblem (see Appendix N) is in terms of the variable $x_{(iu,jv)}^{od}$, some post-processing must be done to determine which variables $x_{(iu,jv)}^{od}$ for arcs $e_a \in ES$ are associated with which missions. After post-processing, for instance, a typical variable such as $x_{(EDAF10.KCHS11)}^{EDARKNGU}$ is identified as the mission leg between airbases EDAF and KCHS of mission number 59. All of the 259 nonzero variables $x_{(iu,jv)}^{od}$ for arcs $e_a \in ES$ from the subproblem solution were post-processed and the results are shown in Appendix O.

All of the twenty commodities considered in the subproblem were delivered resulting in a feasible and optimal solution. Since only cargo which required transshipment was included in the subproblem, all of the cargo needed to be transshipped in the final solution. However, the number of transshipment points varied from one to four. In terms of tonnage, a majority of the cargo (approximately 64 percent) required only one transshipment, approximately 34 percent of the cargo required two transshipments, and approximately 2 percent required three or more transshipments. Based on discussions with the analysts at AMC, cargo is typically transshipped only once. Additionally, the CARGOSIM model assumes only one transshipment (Litko, 22 Sep 92). Therefore, the M^2MCF model may not accurately portray the actual transshipment activity in the AMC channel cargo system, and its solution does not comply with the one-transshipment assumption used by CARGOSIM. However, there may be a way to accurately portray this transshipment activity and allow a maximum of only one transshipment when using the M^2MCF model.

One of the approaches described in Section III.9 for reducing the problem size was to "...allow for commodity-dependent subgraphs..." in the constraint matrix (Helgason and Kennington, 1977:298). The GAMS program, which is presented in Appendix H and was used in the subproblem, does not use this approach. Instead, it replicates the same node-arc incidence matrix for each and every commodity in the constraint matrix. The reason is one of ease and simplicity -- it was easier to generate the subproblem and simpler to develop the FORTRAN code in Appendix H. When implementing the commodity-dependent subgraph approach, only specific mission legs need to be included in the subgraph.

AMC has identified which cargo will require transshipment and what the transshipment points are. For a commodity which must be transshipped, the mission legs which should be represented in the subgraph include those needed to transport the commodity from its origin to its transshipment point and the mission legs needed to transport the commodity from its transshipment point to its destination. All other mission legs (which may cause multiple transshipments) should not be represented in the subgraph. For cargo which can be shipped directly without transshipment, only the mission legs necessary for direct shipment need be represented in the subgraph. Therefore, using the commodity-dependent subgraph approach, the problem size will be reduced and the number of commodities having two or more transshipments may be reduced. However, computational testing has not been done to determine if this approach will reduce the number of transshipments.

Another problem surfaces when the M^2MCF model is used. The problem arises when two conditions exists: (1) There are arcs $e_a \in ES$ (i.e., mission legs) which connect airbase a at time period u to airbase b at time period v and back to airbase a at time period w (where $u \leq v \leq w$); and (2) The unit costs associated with the arcs $e_a \in ES$ are less than the unit costs associated with the arcs $e_a \in ET$, which connect airbase a with itself over time. Since the M^2MCF model attempts to minimize the total transit time, cargo would be routed from node n_{au} to node n_{bv} and then to node n_{aw}. In other words, when the two conditions described above exist, the M^2MCF solution depicts cargo traveling on mission legs rather than having the cargo remain at an airbase for consecutive time periods. This out-and-back phenomena results in the cargo taking up aircraft space unnecessarily, since in the actual channel cargo system, such cargo would remain at the airbase. Additionally, this phenomena incorrectly indicates less transit time than would have been incurred otherwise.

Approximately 20 percent of the cargo in the subproblem was transported on out-and-back mission legs. The commodity-dependent subgraph approach described above may decrease this out-and-back phenomena by eliminating many of the out-and-back mission legs $e_a \in ES$ in the subgraph where the phenomena occurs. If these mission legs are not represented in the subgraph, then the commodity will have to flow on other mission legs or on the arcs $e_a \in ET$. Once again, however, computational testing was not done to determine if this approach would decrease the out-and-back phenomena.

In addition to the assumptions discussed in Chapter I, another assumption must be made when using the M^2MCF formulation. When the channel cargo system is modeled using discrete periods of time (i.e., in the case of the subproblem, eight hour time increments between time periods), the possibility exists that two (or more) aircraft, flying two separate missions, will be departing the same airbase (node n_{iu}) and that the two aircraft will be arriving at another airbase (node n_{jv}) within the same time increment (from time period u to time period v). The result is a network which has two arcs beginning at common origin node n_{iu} and ending at a common terminal node n_{jv}.

There are two ways to model this situation. The first way is to sum the capacities of the two (or more) aircraft and average their respective flight times for that particular mission leg. The sum of the capacities and the average flight time can then represent a pseudo-aircraft which replaces the two aircraft for that mission leg. The second way is to create dummy nodes for each of the two (or more) arcs. The arcs entering and leaving the dummy nodes will have the same capacity as the two aircraft; however, the flight time between the dummy node and either the node n_{iu} or the node n_{jv} will be zero. Although the second method is a more accurate representation of the channel cargo system, the first method was chosen since the first method does not create additional nodes and arcs which would increase the problem size. Therefore, this research assumes that two (or more) aircraft flying between the same two airbases during the same time increment can be modeled by a pseudo-aircraft with combined capacities and average flight times. Since only a small percentage (19 out of 259) of the nonzero variables $x_{(iu,jv)}^{od}$ (for arcs $e_a \in ES$) represented pseudo-aircraft in the subproblem, the assumption that multiple aircraft can be modeled by a pseudo-aircraft may not cause the solution to be much different than if dummy nodes were used to model this situation.

III.13 Additional Comments about the M^2MCF Formulation

The model for the subproblem implemented the idea of steady state conditions as discussed in Section III.4. In other words, there were arcs $e_a \in ET$ which began at time period u=21 and terminated at time period u=1 for all airbases. This formulation has two implications. First, this formulation implies that the routes and schedules are identical from one week to the next. Second, the formulation implies that the cargo generation pattern is the same from one week to the next. Considering an actual monthly schedule generated by AMC, the first implication is not valid since there could be missions which fly only once per month (Robinson, 22 Sep 92). When considering the cargo generation pattern, this formulation is an accurate depiction of the channel cargo system as described in Section III.5.

One of the assumptions discussed in Chapter I was that airbases can handle an unlimited supply of cargo. Limits on this cargo can easily be implemented using the M^2MCF formulation with additional GUB constraints for the arcs $e_a \in ET$. The constraint equation will be identical to Equation (8), except that it will be for all arcs $e_a \in ET$ emanating from airbases with cargo handling or storage area limitations, and the $b_{(iu,iv)}$ parameter would represent the airbase storage capacity.

Because of the limitations described in the previous section, the M^2MCF model is not currently accurate enough to be useful as a scheduling tool. However, the M^2MCF model may still be adequate for AMC advance planning purposes and may be valuable in the two-step, schedule improvement process (described in Section I.3) for improving the monthly flight schedule which is input into CARGOSIM. In addition, as discussed in Chapter IV, the dual variables of the M^2MCF model may provide information which can be used to improve the schedule.

IV. Analysis using the Dual Variables

This chapter details how the dual variables of the multiperiod multicommodity minimal cost flow (M^2MCF) model can provide information as to how the flight schedule can be changed to further minimize the total transit time.

There are two sets of dual variables in the M^2MCF model which may provide information to further decrease the total transit time: the dual variables associated with the Greater Upper Bounding (GUB) constraints, and the dual variables associated with the conservation of flow constraints for the supply nodes (the COF-SN constraints). Each of these two sets of dual variables will be discussed in the following sections.

IV.1 Dual Variables Associated with the GUB Constraints

The dual variables associated with the GUB constraints can be interpreted as the amount by which a one ton increase in the capacity of the aircraft improves, or decreases, the objective function value of the M^2MCF problem (assuming that no other constraints would be violated after the capacity of the aircraft has been increased by one ton). This concept is illustrated with the following example problem shown in Figure 7.

Consider a channel cargo system consisting of three airbases (*a*, *b*, and *c*) and three missions associated with

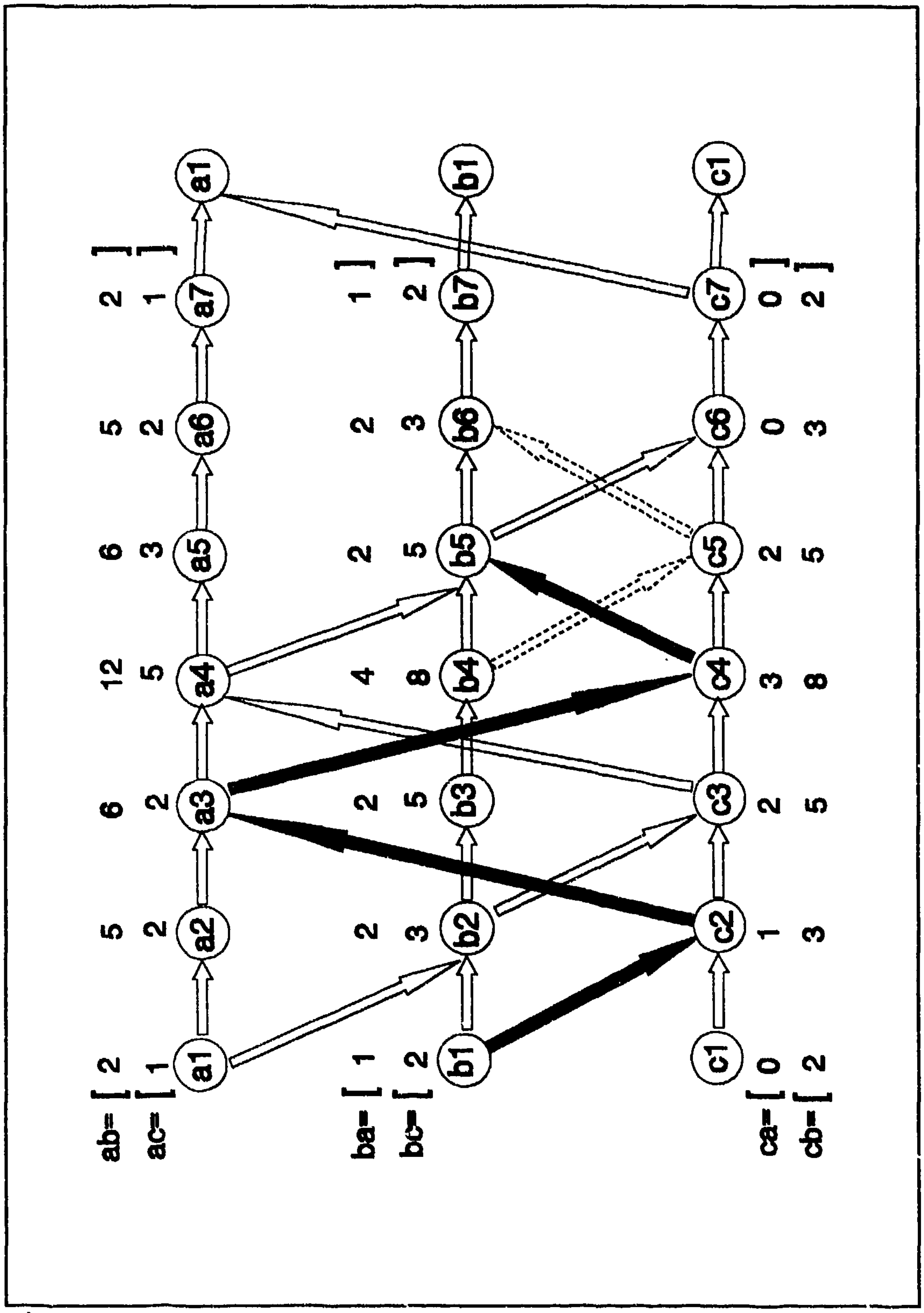

ab= [2 5 6 12 6 5 2]
ac= [1 2 2 5 3 2 1]
ba= [1 2 2 4 2 2 1]
bc= [2 3 5 8 5 3 2]
ca= [0 1 2 3 2 0 0]
cb= [2 3 5 8 5 3 2]
a1 a2 a3 a4 a5 a6 a7 a1
b1 b2 b3 b4 b5 b6 b7 b1
c1 c2 c3 c4 c5 c6 c7 c1

Figure 7

three routes (1, 2, and 3) for a planning horizon of seven days. Mission 1 (shown in Figure 7 as the white arrows connecting different airbases) consists of a C141 aircraft flying route 1: *c-a-b-c* (i.e., from airbase *c* to *a* to *b* and back to *c*). Mission 1 is flown twice in one week departing airbase *c* on day three and on day seven. Mission 2 (shown in Figure 7 as the black arrows) consists of a DC8 aircraft flying route 2: *b-c-a-c-b*. Mission 2 is flown once during the week departing airbase *b* on day one. Mission 3 (shown in Figure 7 as the dotted arrows) consists of a KC10 aircraft flying route 3: *b-c-b*. Mission 3 is also flown only once during the week departing airbase *b* on day four. This channel cargo system has six origin-destination (OD) pairs associated with it: *ab*, *ac*, *ba*, *bc*, *ca*, and *cb*. The cargo generation pattern for each OD pair is shown in Figure 7 in brackets. For example, for OD pair *ab*, two tons arrive at airbase *a* ready to be shipped on day one, five tons arrive on day two, six tons on day three, twelve tons on day four, and so on.

This channel cargo system was modeled using the M^2MCF formulation and solved with a program using the General Algebraic Modeling System (GAMS) language. The capacities of the C141, DC8, and KC10 aircraft are 18, 25, and 30 tons, respectively. For simplicity, all unit cost variables for this example were set equal to one day. The GAMS program is shown in Appendix P. A portion of the results is shown in Appendix Q. An extract of these results showing the objective function value and the marginal costs associated with the GUB constraints are shown in Table 1.

TABLE 1

RESULTS FOR EXAMPLE PROBLEM
(VERSION 1)

OBJECTIVE FUNCTION VALUE: 310.0

CONSTRAINT	MARGINAL COST	CONSTRAINT	MARGINAL COST
A1.B2	-4.000	A3.C4	0.0
A4.B5	-1.000	B1.C2	0.0
B2.C3	0.0	B4.C5	0.0
B5.C6	0.0	C2.A3	0.0
C3.A4	0.0	C4.B5	-1.000
C5.B6	0.0	C7.A1	0.0

Note that the marginal cost of the A1.B2 constraint is -4.0. This implies that a one ton increase in the capacity of the aircraft flying between nodes *a*1 and *b*2 could decrease the overall total transit time by as much as four days.

It is not realistic to increase the capacity of an aircraft. However, dual variables can provide information which justify changing the type of aircraft chosen to service a route. There are three GUB constraints in Table 1 (A1.B2, A4.B5, and C4.B5) whose marginal costs are nonzero.

The zero marginal costs indicate excess capacity on the associated mission legs, while the nonzero marginal costs indicate binding constraints. Since the marginal cost of the A1.B2 constraint is the largest in magnitude compared to the other GUB constraints with a value of -4.0, it appears that the greatest benefit would occur if the route associated with the A1.B2 constraint (route 1) was assigned the DC8 (with a capacity of 25 tons) rather than the C141 (with a capacity of 18 tons). This change of aircraft assignment is also suggested by the A4.B5 constraint which also represents a mission leg in route 1 and has a marginal cost of -1.0. Furthermore, since the C4.B5 constraint (route 2) has a marginal cost of -1.0, it seems logical to assign the KC10 (with a capacity of 30 tons) to that route rather than the DC8. Therefore, the aircraft can be reassigned as follows: the DC8 will fly route 1, the KC10 will fly route 2, and the C141 will fly route 3.

Since the aircraft reassignments will also result in a route (or routes) which has an aircraft with a smaller capacity, the flow along that route will be restricted. And since the absolute value of the dual variable can also be interpreted as the increase in the total transit time per one ton decrease of the aircraft capacity, the objective function value may increase when these aircraft reassignments are made.

As discussed in Chapter II, the marginal cost represents the rate of change in the objective function for a single parameter change (i.e., a change to the right-hand-side value of one constraint). Since we are changing more than one right-hand-side value in the example problem above, exactly how much the objective function will improve will be difficult to determine without resolving the problem. However, we can use the marginal cost associated with the GUB constraints as an upper bound; and therefore, we can conclude that the objective function value could decrease by at most 40 days. (This 40 day decrease is calculated by multiplying the marginal costs by the change in capacity due to the aircraft reassignments and summing all the products: (-4.0)(25 - 18) + (-1.0)(25 - 18) + (-1.0)(30 - 25) = -40.)

Based on this analysis, a second version of the M^2MCF problem was formulated and solved. This time, however, mission 1 consists of a DC8 flying route 1, mission 2 consists of a KC10 flying route 2, and mission 3 consists of a C141 flying route 3. A portion of the results for this second version of the problem is shown in Appendix R. An extract of these results showing the objective function value and the marginal costs associated with the GUB constraints are shown in Table 2. Note that the value of the objective function improved from the 310 days achieved with the first version of the problem (see Table 1) to 294

TABLE 2

RESULTS FOR EXAMPLE PROBLEM
(VERSION 2)

OBJECTIVE FUNCTION VALUE: 294.0

CONSTRAINT	MARGINAL COST	CONSTRAINT	MARGINAL COST
A1.B2	-1.000	A3.C4	0.0
A4.B5	EPS (*)	B1.C2	0.0
B2.C3	0.0	B4.C5	0.0
B5.C6	0.0	C2.A3	0.0
C3.A4	0.0	C4.B5	0.0
C5.B6	0.0	C7.A1	0.0

* EPS means very close to but not equal to zero.

days. Therefore, the change in the aircraft assignment resulted in a decrease of 16 days in the total transit time. This is less than the upper bound of 40 days. Therefore, because of the aircraft reassignments, we can conclude that an aircraft which was previously loaded to capacity now is not or that an aircraft which previously had additional cargo space is now loaded to capacity.

Using the dual information from Table 2, additional aircraft assignment changes may be warranted. Since the A1.B2 constraint (route 1) has a marginal cost of -1.0, it seems reasonable to assign the KC10 (with a capacity of 30 tons) to that route rather than the DC8 (with a capacity of 25 tons). Since all the other marginal costs in Table 2 are equal or nearly equal to zero, no other aircraft assignment changes are suggested. Therefore, a third version of the M^2MCF problem was formulated and solved. This time mission 1 consists of a KC10 flying route 1, mission 2 consists of a DC8 flying route 2, and mission 3 remains the same with a C141 flying route 3. Using the marginal cost as an upper bound, we can conclude that the objective function value could decrease by at most five days.

A portion of the results for this third version of the problem is shown in Appendix S. An extract of these results showing the objective function value and the marginal costs associated with the GUB constraints are shown in Table 3.

TABLE 3

RESULTS FOR EXAMPLE PROBLEM
(VERSION 3)

OBJECTIVE FUNCTION VALUE: 292.0

CONSTRAINT	MARGINAL COST	CONSTRAINT	MARGINAL COST
A1.B2	0.0	A3.C4	0.0
A4.B5	0.0	B1.C2	0.0
B2.C3	0.0	B4.C5	EPS (*)
B5.C6	0.0	C2.A3	0.0
C3.A4	0.0	C4.B5	EPS (*)
C5.B6	0.0	C7.A1	0.0

* EPS means very close to but not equal to zero.

Note that the value of the objective function improved from the 294 days achieved with the second version of this problem (see Table 2) to 292 days. Therefore, the change in the aircraft assignment resulted in a decrease of two days in the objective function value which is less than the upper bound of five days. Additionally, since there were no negative marginal costs associated with the GUB constraints, no further iterations are warranted.

IV.2 Dual Variables Associated with the COF-SN Constraints

The dual variables associated with the COF-SN constraints can be interpreted as the amount by which a one ton decrease in the generation of cargo improves, or decreases, the objective function value of the M^2MCF problem (assuming that no other constraints would be violated after the generation of cargo has been decreased by one ton). To illustrate this concept, the example problem shown in Figure 7 will be used. The aircraft assignments will be the same as the third version of the M^2MCF example problem discussed in the previous section.

A portion of the results is shown in Appendix T. An extract of these results showing the objective function value and the marginal costs associated with the COF-SN constraints are shown in Table 4. These results are from the same solution which is shown in Appendix S and Table 3. Note that the marginal cost associated with the supply node constraint A2.AC is 2.0. This implies that a one ton decrease in the generation of cargo *ac* at node *a*2 will

TABLE 4

RESULTS FOR EXAMPLE PROBLEM
(VERSION 3)

OBJECTIVE FUNCTION VALUE: 292.0

CONSTRAINT	MARGINAL COST	CONSTRAINT	MARGINAL COST
A2.AC	2.000	A3.AB	2.000
A3.AC	1.000	A4.AB	1.000
A4.AC	2.000	A5.AB	4.000
A5.AC	5.000	A6.AB	3.000
A6.AC	4.000	A7.AB	2.000
A7.AC	3.000	B1.BA	2.000
B1.BC	1.000	B2.BA	2.000
B2.BC	1.000	B3.BA	5.000
B3.BC	2.000	B4.BA	4.000
B4.BC	1.000	B5.BA	3.000
B5.BC	1.000	B6.BA	4.000
B6.BC	3.000	B7.BA	3.000
B7.BC	2.000	C1.CA	2.000
C1.CB	4.000	C2.CA	1.000
C2.CB	3.000	C3.CA	1.000
C3.CB	2.000	C4.CA	4.000
C4.CB	1.000	C5.CA	3.000
C5.CB	1.000	C6.CA	2.000
C6.CB	3.000	C7.CA	1.000
C7.CB	2.000		

decrease the overall total transit time for the M^2MCF problem by as much as two days.

The amount of cargo generated on any given day cannot be decreased. However, dual variables can provide information which may justify changing the departure day of the mission. Since the marginal cost of the A5.AC and the B3.BA constraints in Table 4 are the largest in magnitude compared to the marginal costs of the other COF-SN constraints, it appears that a benefit would occur if the departure day of a mission was changed so that the cargo *ac* could depart node *a*5 and arrive at any node *c* in a quicker manner than the current network permits. Likewise, it appears that an additional benefit would occur if the departure day of a mission was changed so that the cargo *ba* could depart node *b*3 and arrive at any node *a* in a quicker manner than the current network permits. How much of a benefit will be obtained cannot be determined from the dual variables since the mission departure time, and not the cargo generation, is the factor being affected.

There are several ways to change the mission departure times of the network based on the information from the dual variables. One way to change the network, based on the information from the marginal cost of the B3.BA constraint, is to change mission 2 from departing on day one to departing on day three. This network change is shown in Figure 8. This change enables commodity *ba*, which is generated on day three, to arrive at airbase *a* on day five using mission 3. Therefore, the network change enables commodity *ba* to arrive at a node *a* in a quicker manner than the previous network allowed. Another way to change the network, based on the information from the marginal cost of the A5.AC constraint, is to change mission 1 from departing on days three and seven to departing on days one and four.

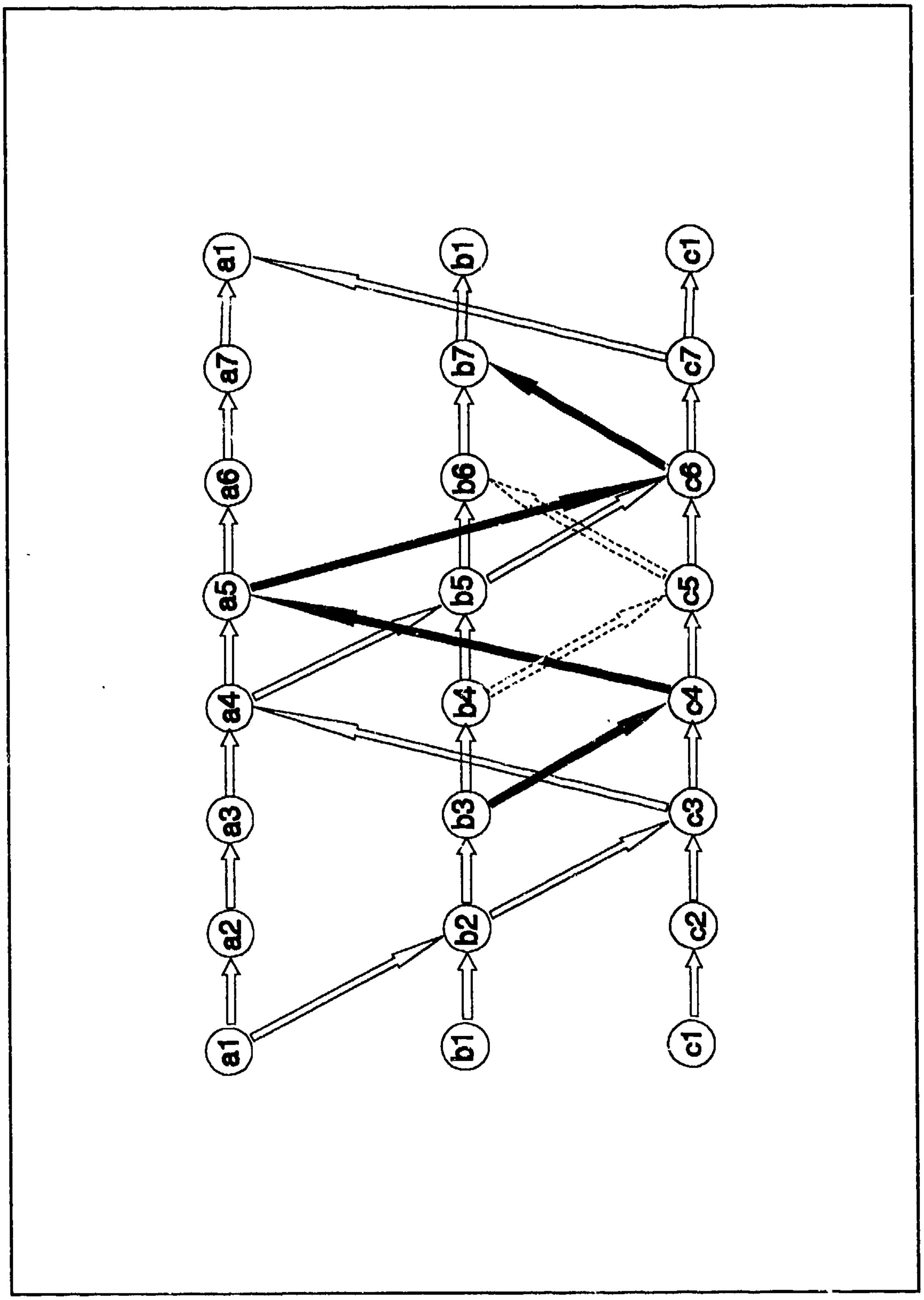


Figure 8

This network change is shown in Figure 9. This change enables commodity AC, which is generated on day five, to arrive at airbase *c* on day seven using mission 1.

There may be several other ways to change the network based on the marginal costs of these two constraints. The two network changes presented above may not be the most effective changes possible and were chosen only as examples of possible changes.

Based on the network shown in Figure 8, a fourth version of the M^2MCF problem was formulated. A portion of the results is shown in Appendix U. An extract of these results showing the objective function value and the marginal costs associated with the COF-SN constraints are shown in Table 5. Note that the objective function value decreased from 292 days in the third version (See Table 4) to 273 days in the fourth version.

Based on the network shown in Figure 9, a fifth version of the M^2MCF problem was formulated. A portion of the results of this fifth version of the problem is shown in Appendix V. An extract of these results showing the objective function value and the marginal costs associated with the COF-SN constraints are shown in Table 6. Note that the objective function value decreased from 292 days in the third version (See Table 4) to 278 days in the fifth version.

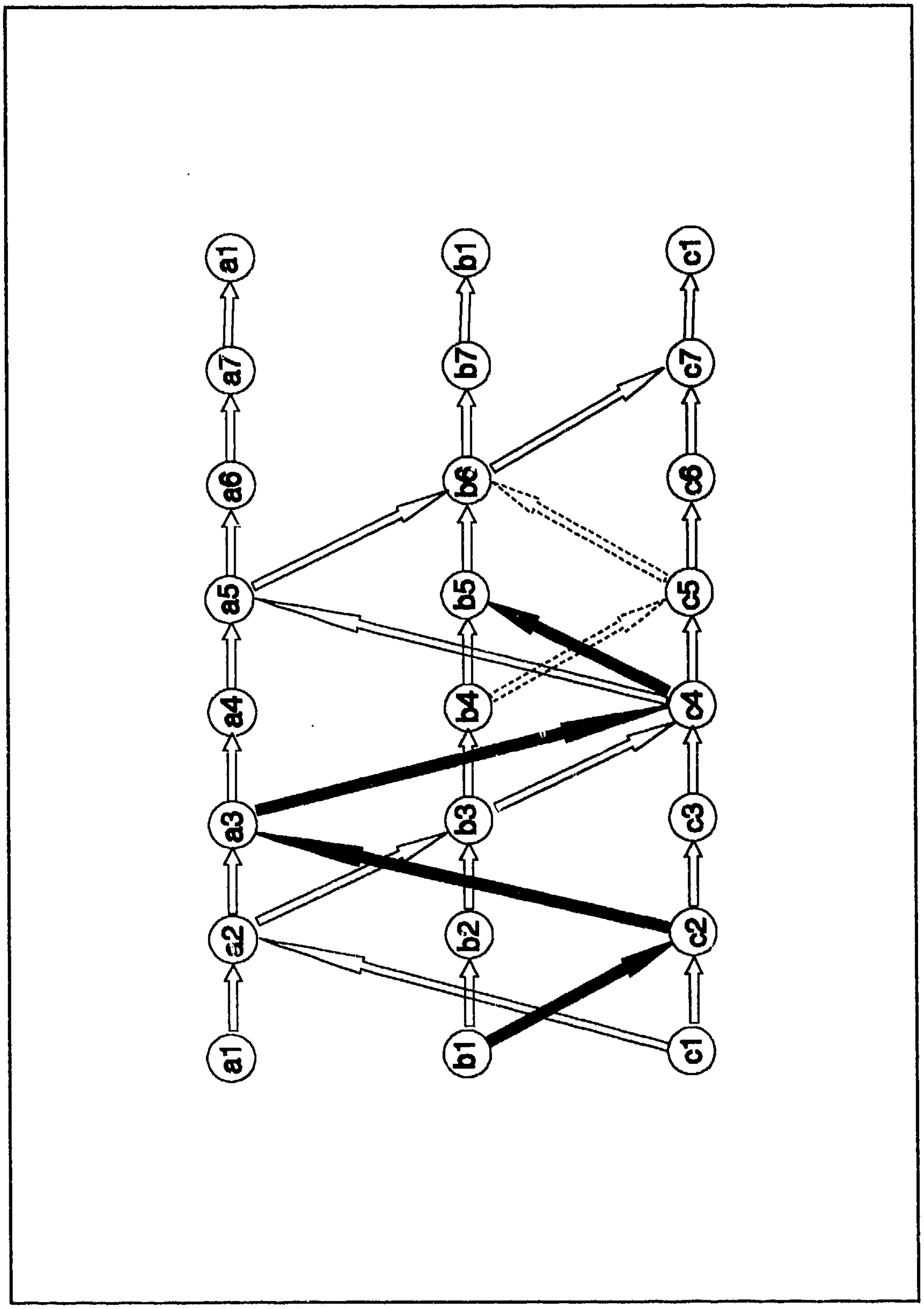


Figure 9

TABLE 5

RESULTS FOR EXAMPLE PROBLEM
(VERSION 4)

OBJECTIVE FUNCTION VALUE: 273.0

CONSTRAINT	MARGINAL COST	CONSTRAINT	MARGINAL COST
A2.AC	4.000	A3.AB	3.000
A3.AC	3.000	A4.AB	2.000
A4.AC	2.000	A5.AB	2.000
A5.AC	1.000	A6.AB	3.000
A6.AC	4.000	A7.AB	2.000
A7.AC	3.000	B1.BA	3.000
B1.BC	3.000	B2.BA	2.000
B2.BC	2.000	B3.BA	2.000
B3.BC	2.000	B4.BA	4.000
B4.BC	2.000	B5.BA	3.000
B5.BC	2.000	B6.BA	5.000
B6.BC	5.000	B7.BA	4.000
B7.BC	4.000	C1.CA	3.000
C1.CB	5.000	C2.CA	2.000
C2.CB	4.000	C3.CA	1.000
C3.CB	3.000	C4.CA	1.000
C4.CB	2.000	C5.CA	3.000
C5.CB	1.000	C6.CA	2.000
C6.CB	1.000	C7.CA	1.000
C7.CB	2.000		

Using the dual information from this fifth version of the problem (see Table 6), other changes to mission departure times may be warranted. Since the marginal cost of the A6.AC and the B4.BA constraints are the largest in magnitude compared to the marginal costs of the other COF-SN constraints, mission departure times can be changed based on these two marginal costs. Instead of analyzing both of these possibilities, only a mission departure time based on

TABLE 6

RESULTS FOR EXAMPLE PROBLEM
(VERSION 5)

OBJECTIVE FUNCTION VALUE: 278.0

CONSTRAINT	MARGINAL COST	CONSTRAINT	MARGINAL COST
A2.AC	2.000	A3.AB	2.000
A3.AC	1.000	A4.AB	2.000
A4.AC	3.000	A5.AB	1.000
A5.AC	2.000	A6.AB	4.000
A6.AC	5.000	A7.AB	3.000
A7.AC	4.000	B1.BA	2.000
B1.BC	1.000	B2.BA	3.000
B2.BC	2.000	B3.BA	2.000
B3.BC	1.000	B4.BA	5.000
B4.BC	1.000	B5.BA	4.000
B5.BC	2.000	B6.BA	3.000
B6.BC	1.000	B7.BA	3.000
B7.BC	2.000	C1.CA	1.000
C1.CB	2.000	C2.CA	1.000
C2.CB	3.000	C3.CA	2.000
C3.CB	2.000	C4.CA	1.000
C4.CB	1.000	C5.CA	4.000
C5.CB	1.000	C6.CA	3.000
C6.CB	4.000	C7.CA	2.000
C7.CB	3.000		

the B4.BA constraint has been considered. Once again, there are several ways to change the network based on the marginal cost of the B4.BA constraint. Only one change, chosen as an example, has been considered.

The change to the network, based on the information from the marginal cost of the B4.BA constraint, is to change mission 2 from departing on day three as shown in Figure 8 to departing on day four as shown in Figure 10. This change

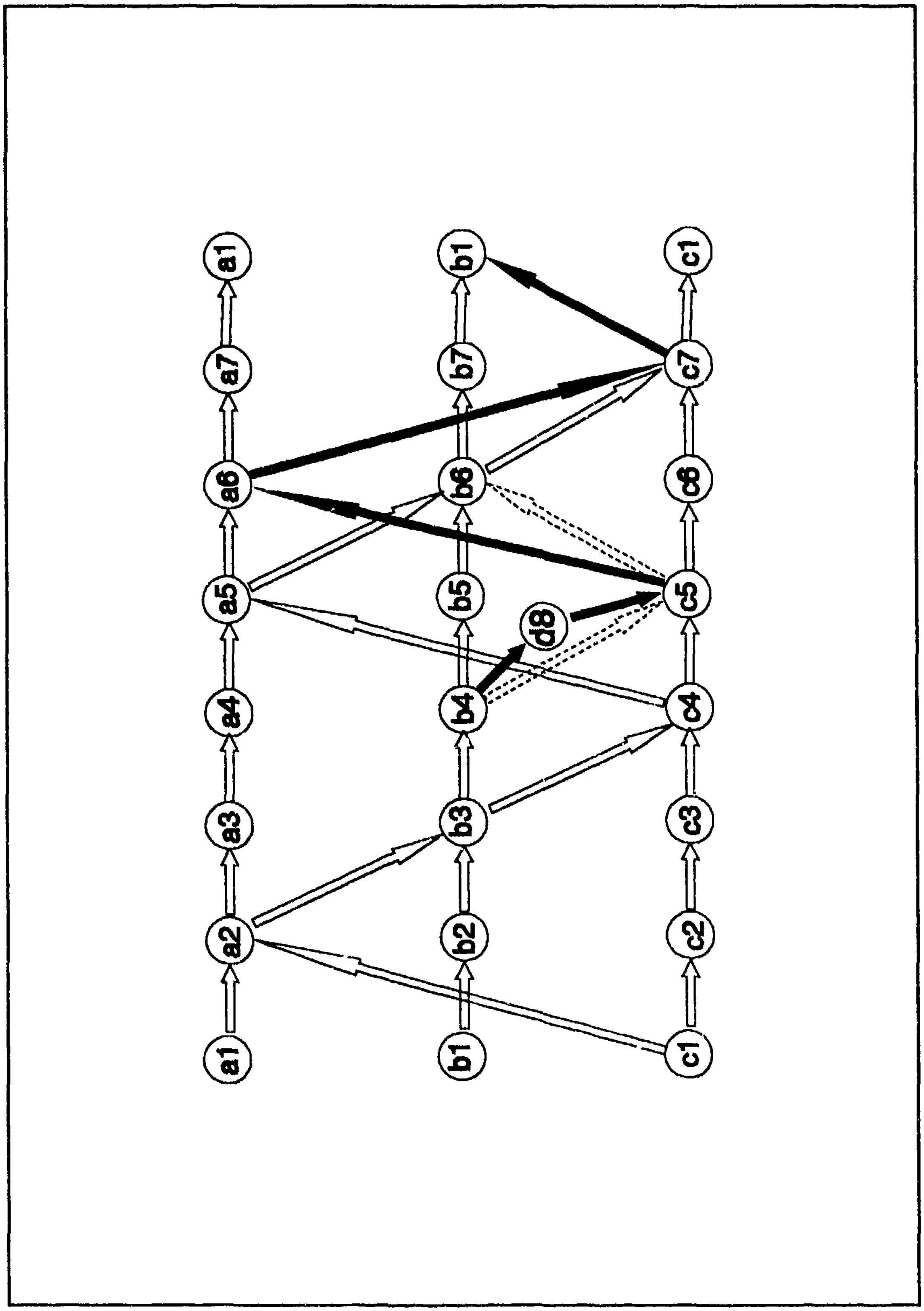

a1
a2
a3
a4
a5
a6
a7
a1
b1
b2
b3
b4
b5
b6
b7
b1
d8
c1
c2
c3
c4
c5
c6
c7
c1

Figure 10

enables commodity *ba*, which is generated on day four to arrive at airbase *a* on day six using mission 2. Note that since routes 2 and 3 overlap from *b*4 to *c*5, route 2 is modeled using a dummy node, *d*8, to distinguish between the two different mission legs. The GAMS program for this sixth version of the M^2MCF problem is shown in Appendix W. A portion of the results is shown in Appendix X. An extract of these results showing the objective function value and the marginal costs associated with the COF-SN constraints are shown in Table 7. Note that the objective function value decreased from 278 days in the fifth version (See Table 6) to 274 days in this sixth version.

Once again, using the dual information from this sixth version of the problem (see Table 7), other changes to mission departure times may be warranted. Since the marginal cost of the B7.BA constraint is the largest in magnitude, with a value of 5.0, compared to the marginal costs of the other COF-SN constraints, mission departure times can be changed based on this marginal cost. Only one change to the network, chosen as an example, has been considered based on the marginal cost of the B7.BA constraint. Mission 1 is changed from departing on days one and four as shown in Figure 10 to departing on days one and five as shown in Figure 11. This change enables commodity *ba*, which is generated on day seven, to arrive at airbase *a*

TABLE 7

RESULTS FOR EXAMPLE PROBLEM
(VERSION 6)

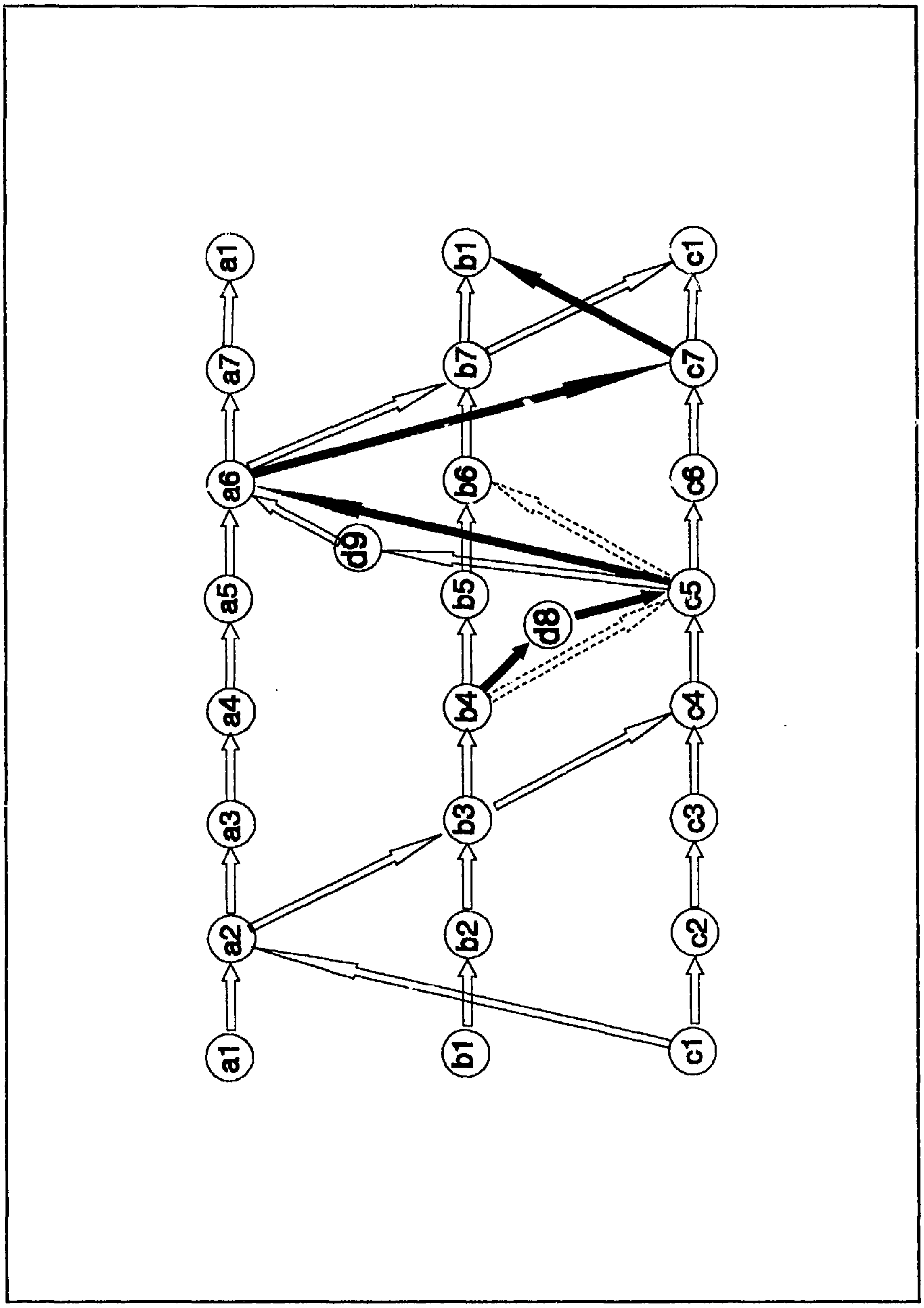
OBJECTIVE FUNCTION VALUE: 274.0

CONSTRAINT	MARGINAL COST	CONSTRAINT	MARGINAL COST
A2.AC	2.000	A3.AB	3.000
A3.AC	4.000	A4.AB	2.000
A4.AC	3.000	A5.AB	1.000
A5.AC	2.000	A6.AB	2.000
A6.AC	1.000	A7.AB	3.000
A7.AC	4.000	B1.BA	4.000
B1.BC	3.000	B2.BA	3.000
B2.BC	2.000	B3.BA	2.000
B3.BC	1.000	B4.BA	2.000
B4.BC	1.000	B5.BA	4.000
B5.BC	2.000	B6.BA	3.000
B6.BC	1.000	B7.BA	5.000
B7.BC	4.000	C1.CA	-2.000
C1.CB	2.000	C2.CA	3.000
C2.CB	4.000	C3.CA	2.000
C3.CB	3.000	C4.CA	1.000
C4.CB	2.000	C5.CA	1.000
C5.CB	1.000	C6.CA	EPS (*)
C6.CB	2.000	C7.CA	-1.000
C7.CB	1.000		

* EPS means very close to but not equal to zero.

on day two using mission 1. Note again, that dummy nodes, *d*8 and *d*9, are used to distinguish mission legs where the legs share a common origin node and a common terminal node.

This seventh version of the problem was solved, and a portion of the results is shown in Appendix Y. An extract of these results showing the objective function value and the marginal costs associated with the COF-SN constraints


a1
a2
a3
a4
a5
a6
a7
a1
b1
b2
b3
b4
b5
b6
b7
b1
c1
c2
c3
c4
c5
c6
c7
c1
d8
d9
Figure 11

are shown in Table 8. This time, however, the objective function value increased from 274 days in the sixth version (See Table 7) to 308 days in this seventh version. Additionally, several COF-SN constraints have marginal costs which equal or exceed the value of 5.0. Therefore, the last change to the mission departure time worsened the overall total transit time. Thus, one can conclude that this process is not guaranteed to result in an improvement to the objective function value.

IV.3 Chapter Summary

This chapter demonstrated that the dual variables of the M^2MCF problem can be useful in examining ways to improve the network and schedule. The dual information can be used to change either the type of aircraft assigned to a particular route or the departure time of the mission. However, such changes are not guaranteed to improve the objective function value. Since any change to the network or schedule typically involves changing several parameters at once, the exact impact of a change is difficult to determine but can be evaluated by solving the changed network.

TABLE 8

RESULTS FOR EXAMPLE PROBLEM
(VERSION 7)

OBJECTIVE FUNCTION VALUE: 308.0

CONSTRAINT	MARGINAL COST	CONSTRAINT	MARGINAL COST
A2.AC	2.000	A3.AB	5.000
A3.AC	4.000	A4.AB	4.000
A4.AC	3.000	A5.AB	3.000
A5.AC	2.000	A6.AB	2.000
A6.AC	1.000	A7.AB	3.000
A7.AC	4.000	B1.BA	5.000
B1.BC	3.000	B2.BA	4.000
B2.BC	2.000	B3.BA	3.000
B3.BC	1.000	B4.BA	2.000
B4.BC	1.000	B5.BA	4.000
B5.BC	3.000	B6.BA	3.000
B6.BC	2.000	B7.BA	2.000
B7.BC	1.000	C1.CA	1.000
C1.CB	2.000	C2.CA	4.000
C2.CB	6.000	C3.CA	3.000
C3.CB	5.000	C4.CA	2.000
C4.CB	4.000	C5.CA	1.000
C5.CB	3.000	C6.CA	EPS (*)
C6.CB	2.000	C7.CA	-1.000
C7.CB	1.000		

* EPS means very close to but not equal to zero.

V. Conclusions and Recommendations

V.1 Conclusions

The purpose of this research has been to develop an algorithm which, given a flight schedule and cargo requirements, determines a flow of cargo between OD pairs which minimizes the cargo's delay enroute. This research shows that the AMC channel cargo system can be modeled using a multiperiod multicommodity minimal cost flow (M^2MCF) model. The objective of this model is to minimize the total transit time for all commodities. Additionally, if the missions and cargo generation are the same from one planning period to the next, then the network representing the channel cargo system can be modified to represent this steady state condition. However, there are unresolved problems with the presented model that limit the applicability of the model. Currently, the model is not accurate enough to be useful as a scheduling tool, but it may be adequate for AMC advance planning purposes.

There are several advantages to modeling the channel cargo system using an M^2MCF model. First, this model accounts for the two types of delay enroute: the delay caused when cargo is at the origin base awaiting transportation, and the delay incurred after cargo has left the origin base (where the latter type of delay enroute includes the flight time and the time that cargo waits for transportation at a transshipment point). Another advantage of using the M^2MCF model is that it uses the same information that AMC's STORM and CARGOSIM models use. Therefore, it is compatible with AMC's current scheduling process. Additionally, it would be easy to model any cargo handling and capacity restrictions at an airbase by adding additional Greater Upper Bounding constraints to the M^2MCF formulation. Furthermore, there is another advantage when the channel cargo system is modeled using the M^2MCF formulation with steady state conditions. When the analysts at AMC use CARGOSIM, they must replicate the monthly flight schedule three times. The purpose of the first of these schedules is to simulate the backlog of cargo which is awaiting transportation for the next month. The M^2MCF model with steady state conditions also performs this same function by returning undelivered cargo to the beginning of the planning period.

There are, however, limitations to modeling the channel cargo system using an M^2MCF formulation. Since the channel cargo system has a large number of commodities and missions associated with it, the size of the M^2MCF model of the entire system is larger than what AMC's current automation system is capable of solving. Therefore, the problem size must be reduced using the approaches described in Chapter

III. Unfortunately, one of these approaches for reducing the problem size (decreasing the number of time periods) has the effect of creating a less accurate representation of the channel cargo system. When the M^2MCF model is used in conjunction with steady state conditions, more inaccuracies are created. The steady state conditions assume that missions are repetitive from one planning period to the next. For example, using a planning horizon of one week and modeling a mission which flies once in that week implies that the mission is flown four times in a month. Therefore, when using a one week planning horizon, a mission that is flown only once a month cannot be accurately represented.

Furthermore, the M^2MCF model does not comply with a major assumption used by the AMC analysts and CARGOSIM. This assumption is that only one transshipment may occur when cargo is delivered. More than one transshipment, however, can occur when the M^2MCF model is used. Additionally, when the two conditions described in Section III.12 exist, the M^2MCF depicts cargo traveling on mission legs from a particular airbase to another airbase and back to the particular airbase rather than having the cargo remain at the particular airbase for consecutive time periods. This out-and-back phenomena results in cargo taking up aircraft space unnecessarily, since in the actual channel cargo system, such cargo would have remained at the particular airbase.

Because of these limitations, the M^2MCF model is currently not a good tool to determine real time cargo flow. However, the M^2MCF model may still be valuable. As discussed in Section I.3, a two-step, iterative process was proposed to improve the monthly flight schedule generated by STORM and CARGPREP prior to its input into CARGOSIM. The M^2MCF model may be adequate to serve as the first step of this iterative, schedule improvement process. Furthermore, as discussed in Chapter IV, the dual variables of the M^2MCF model may provide useful information to improve AMC's monthly flight schedule.

V.2 Recommendations

There are five areas where future research is recommended: (1) Correcting the problems identified in Section V.1; (2) Researching the effects of and solutions to decomposing the channel cargo system into four geographic areas; (3) Developing the process to improve the flight schedule based on cargo flow; (4) Developing and testing the two-step schedule improvement process described in Chapter I; and (5) Refining the M^2MCF model and portraying more accurately the channel cargo system.

One method which may correct the problems identified in Section V.1 is to implement the approach described in

Section III.9 which was not used in this research. This approach is to allow for commodity-dependent subgraphs when formulating the constraint matrix of the M^2MCF model. In other words, for a given commodity, the specific mission legs which can deliver this commodity should be determined. Then only the arcs representing these mission legs should be included in the node-arc incidence matrix associated with the commodity. As discussed in Chapter III, the commodity-dependent subgraph approach will decrease the M^2MCF problem size, may decrease the amount of cargo having two or more transshipments, and may decrease the out-and-back phenomena described above. Therefore, future research can investigate ways to efficiently derive these commodity-dependent subgraphs and evaluate the impact of this approach by performing computational tests.

Since the problem size is too large when modeling the entire channel cargo system, it was suggested in Section III.9 that the system be broken down into four geographic areas and that four subproblems representing these areas be solved. However, this decomposition method would result in four independent solutions, when in fact, there exists interdependence between the four areas. For example, there are OD pairs which link the European/Southwest Asia (E/SWA) area to the Africa area and to the Pacific area, and there are routes which connect these different areas (Robinson, 22

Sep 92). Therefore, it is recommended that future research investigate the effects of this decomposition and determine ways to account for the interdependency between geographic areas. One technique to account for this interdependency is to solve a subproblem for one of the areas and use the information on the amount of cargo shipped when solving another subproblem. For example, when solving a subproblem for the E/SWA area, if a particular mission (which connects the E/SWA area to the Pacific area) is used to transport cargo, then information on the amount of cargo transported is used when solving the Pacific area subproblem. One way to use this information is to subtract the amount of cargo determined in the E/SWA subproblem from the capacity of the aircraft flying the mission in the Pacific subproblem. This has the effect of coordinating the delivery of cargo between the two geographic areas by ensuring that the capacity of the aircraft servicing both areas is not exceeded.

In Chapter I, a two-step iterative process was suggested to improve the monthly flight schedule which AMC inputs into CARGOSIM. The first step of this process is to determine cargo flow given a monthly flight schedule, while the second step is to modify the schedule based on the cargo flow. The M^2MCF model may be adequate to use as the first step of this schedule improvement process. However, further research is needed to develop the second step of this process and to improve the flight schedule based on cargo flow. One method which may improve the flight schedule was developed by Captain Gregory S. Rau (Rau, 1993). Another method which may improve the schedule was presented in Chapter IV. In that chapter, it was demonstrated that the dual variables may provide information to modify the schedule and further minimize the total transit time. This dual information can be used to change either the type of aircraft assigned to a particular route or the departure time of the mission. Further research may develop an algorithm or heuristic which uses the dual information to improve the monthly flight schedule.

The primary purpose of this research was to develop a cargo flow approach that would be a major component of a scheduling algorithm for AMC advance planning. The basic foundation work has been done for this algorithm (Rau, 1993), and the next logical step is to develop and test this schedule improvement process.

Finally, further research is recommended to refine the M^2MCF model and to portray more accurately the channel cargo system. For instance, as stated in Section III.12, two or more aircraft which departed an airbase and arrived at a common airbase within the same time increment were modeled as a pseudo-aircraft with a combined capacity and an average flight time. Further research can be done to determine if a weighted average for the flight times is significantly more accurate. Additionally, the alternative approach to modeling this phenomenon by using dummy nodes (as demonstrated in Chapter IV) instead of pseudo-aircraft could be implemented and tested. Furthermore, one of the assumptions made in Section I.5 was that cargo was classified by weight only and considered generic in all other respects (i.e., no priority considerations). Future research can investigate the relaxation of this assumption. One approach is to assign higher values to the unit costs of high priority cargo. For instance, in reality a particular mission flight time may be four hours. However, the unit cost for that mission for a high priority cargo could be set equal to a higher value (i.e., eight hours).

Additionally, as described in Section III.13, there are two implications when the M^2MCF model is used with steady state conditions: routes and schedules are the same from one planning period to the next; and the cargo generation pattern is the same from one planning period to the next. Future research is recommended to examine the significance of these implications and test the impact of using the model with steady state conditions. Finally, another assumption made in Section I.5 was that maximizing the cargo load of each aircraft was of secondary importance to minimizing the delay enroute. AMC is actually concerned with both of these goals. Therefore, future research could look into methods to satisfy both goals. Techniques which could accommodate these goals, such as goal programming, could be considered.

PAGES APPENDIXS A-B-C-D

ARE

MISSING

IN

ORIGINAL

DOCUMENT

Appendix E: Cargo Generation for Subproblem

This appendix contains the cumulative amounts of the commodities which arrive during a one week period. This data was obtained from the "demand.raw" file of a recent AMC study (Robinson, 22 Sep 92) and used as input data for the subproblem in this research. The first two columns in the table show the OD pair using the ICAO codes. The remaining columns show the cumulative tonnage of cargo which arrives at the origin base for each day of the week beginning on Friday and ending on Thursday.

EDAR	KNGU	0.24	0.48	0.72	0.96	1.20	1.44	1.68
EDAR	LGIR	0.30	0.59	0.89	1.19	1.48	1.78	2.08
EDAR	LICZ	0.18	0.36	0.54	0.72	0.90	1.08	1.26
EDAR	LIRN	0.18	0.37	0.55	0.73	0.92	1.10	1.28
EDAR	OEDR	0.85	1.69	2.54	3.39	4.23	5.08	5.93
EGUN	KNGU	0.78	1.56	2.34	3.12	3.90	4.68	5.46
EGUN	LTAG	1.68	3.36	5.04	6.72	8.40	10.08	11.76
KCHS	EDAF	0.16	0.20	0.22	0.46	0.75	1.01	1.24
KDOV	LGIR	0.31	0.37	0.37	0.73	1.15	1.64	2.12
KDOV	LIPA	6.24	7.32	7.50	14.65	23.05	32.91	42.58
KDOV	OEDR	6.26	7.35	7.53	14.70	23.14	33.04	42.75
KNGU	LIPA	1.19	1.74	2.01	3.95	6.00	8.32	10.50
KTIK	LGIR	0.24	0.36	0.43	0.68	1.09	1.48	1.87
KTIK	LIPA	0.51	0.77	0.91	1.45	2.30	3.12	3.94
KTIK	LTAG	0.83	1.24	1.47	2.35	3.73	5.06	6.39
KTIK	OEDR	0.94	1.41	1.67	2.65	4.22	5.72	7.23
KTIK	OERY	0.50	0.75	0.89	1.42	2.26	3.07	3.87
LETO	KDOV	8.19	16.37	24.56	32.75	40.93	49.12	57.31
LETO	KTIK	0.77	1.54	2.31	3.08	3.85	4.62	5.39
LETO	KWRI	1.16	2.32	3.48	4.64	5.80	6.96	8.12

Appendix F: Airbases for Subproblem

This appendix contains the ICAO codes for the airbases used as input data for the subproblem in this research. The data was obtained from the "base.dat" file of a recent AMC study (Robinson, 22 Sep 92).

BIKF
CYQX
EDAF
EDAR
EGUN
EXXX
FTTJ
FZAA
GLRB
GOOY
HKNA
HSSS
KCHS
KDOV
KNGU
KSUU
KTIK
KWRI
KXXX
LCRA
LERT
LETO
LGIR
LICZ
LIPA
LIRN
LIRP
LLBG
LPLA
LTAG
OBBI
OEDR
OERY
OJAF
OKBK
OMFJ

Appendix G: Routes for Subproblem

This appendix contains the routes used as input data for the subproblem in this research. The data was obtained from the "route.dat" and "planes.out" files of a recent AMC study (Robinson, 22 Sep 92). The first column contains the route number. The subsequent columns outline the specific route using the four-letter ICAO code for each stop and a code number to designate the reason for the stop.

```
  3 EXXX1 KTIK4 CYQX4 EDAR4 EXXX9
 56 KSUU1 KTIK4 KDOV6 EDAF6 KDOV6 KTIK4 KSUU9
 58 KSUU1 KTIK4 KDOV6 EDAR6 KDOV6 KTIK4 KSUU9
 59 KSUU1 KTIK4 KDOV6 EGUN6 EDAR4 EDAF6 KCHS6 KTIK4 KSUU9
137 KXXX1 KTIK4 EDAF4 KDOV4 KTIK4 KXXX9
180 KDOV1 EDAF6 KDOV9
181 KDOV1 EDAR6 KDOV9
196 KCHS1 KNGU4 LPLA6 GOOY6 GLRB4 FZAA6 FTTJ4 FZAA6 GOOY4
    LPLA6 KNGU4 KCHS9
200 KDOV1 EDAR6 OJAF6 EDAR6 KDOV9
202 KCHS1 KNGU4 BIKF6 EGUN4 KCHS9
203 KDOV1 KCHS4 KNGU4 BIKF6 EGUN4 KDOV9
216 KCHS1 KNGU4 LERT6 LICZ4 OBBI4 OMFJ6 OBBI4 LICZ6 LERT4
    LPLA6 KNGU4 KCHS9
224 KDOV1 EDAF6 OEDR4 EDAF6 KDOV9
225 KSUU1 KTIK4 KWRI6 LPLA4 EDAF6 KWRI6 KTIK4 KSUU9
230 EDAF1 LETO4 LIPA6 EDAR4 EGUN4 EDAF9
231 EDAF1 EGUN4 EDAR6 LIPA4 LETO4 EDAF9
235 EDAF1 OKBK4 OEDR6 OERY4 EDAF9
237 EDAF1 LTAG4 EDAF9
239 EDAR1 LTAG4 EDAR9
241 KDOV1 LETO6 KDOV9
242 KWRI1 LPLA6 KWRI9
249 EGUN1 EDAR4 [illegible] LIPA6 LETO4 EDAR4 EGUN9
251 EGUN1 EDAF4 LIPA6 LGIR4 LCRA4 LTAG6 LCRA4 LGIR4 LIPA6
    EDAF4 EGUN9
252 KDOV1 EDAR4 LTAG4 EDAR4 KDOV9
255 KDOV1 KNGU4 LERT6 OBBI4 LICZ6 LERT6 KNGU4 KDOV9
259 KCHS1 KNGU4 LERT6 LIRN4 LICZ6 LIRN4 LERT6 KNGU4 KCHS9
260 KCHS1 KNGU4 LERT6 LIRN4 LERT6 KNGU4 KCHS9
262 EDAF1 EGUN4 [illegible] LIPA4 LETO4 EDAF4 LTAG6 EDAF4 LETO4
    LIPA4 EDAR4 EGUN4 EDAF9
264 EDAF1 LIRN4 LICZ4 LERT6 LICZ4 LIRN4 EDAF9
265 KCHS1 KNGU4 LERT6 LIRN4 LICZ4 OBBI6 OMFJ4 OBBI4 LICZ6
    LIRN4 LERT6 [illegible] KNGU4 KCHS9
266 EDAF1 LIRN4 LICZ4 LIRN4 EDAF9
269 KDOV1 EDAF4 OERY6 EDAF4 KDOV9
270 KWRI1 LPLA4 EDAR6 LPLA4 KWRI9
```

```
271 EDAF1 OEDR6 EDAF9
292 EDAF1 EDAR4 EDAF9
293 KDOV1 EDAR4 LLBG4 EDAR4 KDOV9
294 KNGU1 LETO4 LICZ4 HSSS4 HKNA4 LICZ4 LPLA4 KNGU9
```

Appendix H: Schedule for Subproblem

This appendix contains an extract of the information used to develop the schedule for the subproblem in this research. The data was obtained from the "schedule.raw" file of a recent AMC study (Robinson, 23 Sep 92). The first column contains the route number, the second column contains the aircraft type selected for that route, and the third column contains the day that the aircraft departs the base (decimals are used to indicate the "time" of day that the aircraft departed).

19	C005	0.1
19	C005	15.1
23	C005	1.2
37	C005	2.3
56	C005	3.4
58	C005	4.5
58	C005	12.0
58	C005	19.5
58	C005	27.0
60	C005	5.6
	.	
	.	
	.	
	.	
	.	
252	KC10	12.5
252	KC10	14.8
252	KC10	17.1
252	KC10	19.5
252	KC10	21.8
252	KC10	24.1
252	KC10	26.4
252	KC10	28.7
252	KC10	1.0
253	KC10	4.4

Appendix I: Flight Data for Subproblem

This appendix contains an extract of the flight times between airbases used as input data for the subproblem in this research. The data was obtained from the "fly.dat" file of a recent AMC study (Robinson, 22 Sep 92). The first column contains the ICAO codes for the starting airbase of a mission leg, the second column contains the ICAO codes for the ending airbase of a mission leg, and the third through the ninth columns contain the flight times (in hours) between the two airbases for the various aircraft types. The fourth columns contains the flight times for a C141 aircraft. AMC actually only uses the fourth column in the table to calculate flight times for the other various aircraft types by using a multiplication factor in the "jet.dat" of their recent study.

ABAS	ASRI	2.7	2.7	2.7	2.7	2.7	2.7	2.7
APLM	ASRI	4.7	4.7	4.7	4.7	4.7	4.7	4.7
APWR	ASRI	1.8	1.8	1.8	1.8	1.8	1.8	1.8
ASRI	ABAS	3.0	3.0	3.0	3.0	3.0	3.0	3.0
ASRI	APLM	5.8	5.8	5.8	5.8	5.8	5.8	5.8
ASRI	APWR	2.2	2.2	2.2	2.2	2.2	2.2	2.2
ASRI	NSTU	5.5	5.5	5.5	5.5	5.5	5.5	5.5
ASRI	NZCH	3.0	3.0	3.0	3.0	3.0	3.0	3.0
BGSF	BGTL	1.8	1.8	1.8	1.8	1.8	1.8	1.8
BGSF	CYYR	2.7	2.7	2.7	2.7	2.7	2.7	2.7
	.							
	.							
	.							
	.							
	.							
KSUU	KRIV	1.9	1.9	1.9	1.9	1.9	1.9	1.9
KSUU	PADK	6.5	6.5	6.5	6.5	6.5	6.5	6.5
LERT	OBBI	7.0	7.0	7.0	7.0	7.0	7.0	7.0
PGUA	RJTY	4.2	4.2	4.2	4.2	4.2	4.2	4.2
PHIK	PWAK	5.4	5.4	5.4	5.4	5.4	5.4	5.4
PHIK	RODN	10.2	10.2	10.2	10.2	10.2	10.2	10.2
RODN	WSAP	5.6	5.6	5.6	5.6	5.6	5.6	5.6
RPMB	WIIH	4.2	4.2	4.2	4.2	4.2	4.2	4.2
WIIH	RPMB	4.4	4.4	4.4	4.4	4.4	4.4	4.4
WSAP	RODN	5.1	5.1	5.1	5.1	5.1	5.1	5.1

Appendix J: GAMS.FOR Program

This appendix contains the FORTRAN program, "GAMS.FOR", used to create the GAMS program for the subproblem in this research. The GAMS program is shown in Appendix K.

```
      PROGRAM WRITEGAMS
C
C   AC(*) = AIRCRAFT TYPE FOR OCCURANCE * OF CURRENT ROUTE
C   ARRCH = CHARACTER FORM OF ARRCON
C   ARRCON = CONVERTED LEG ARR. PERIOD FOR CURRENT OCCUR OF
C            CURRENT RTE
C   ARRTIM = LEG ARRIVAL TIME FOR CURRENT OCCURANCE OF CURRENT
C            ROUTE
C   AVGFLT = AVERAGE FLT TIME ACROSS IDENTICAL LEGS = (CUMFLT /
C            COUNTER)
C   CAP(*) = CUM. CAPACITY OF AIRCRAFT FLYING GIVEN LEG OF A
C            MISSION
C   CAPAC = CAPACITY OF SPECIFIC AIRCRAFT FLYING GIVEN MISSION
C   CNT22 = # OF LINES (ENTRIES) IN TEMP. FILE #4 (UNIT=22)
C   COUNT = COUNTS NUMBER OF OCCURENCES OF IDENTICAL LEGS
C   CUMDEM(*,?) = CUMULATIVE DEMAND FOR THE WEEK AS OF DAY ? FOR
C                 ROW *
C   DEPART(*) = ORIG. DEPARTURE TIME FOR OCCURANCE * OF CURRENT
C               ROUTE
C   DEPCH = CHARACTER FORM OF DEPCON
C   DEPCON = CONVERTED LEG DEP. PERIOD FOR CURRENT OCCUR OF
C            CURRENT RTE
C   DEPTIM = LEG DEPARTURE TIME FOR CURRENT OCCURANCE OF CURRENT
C            ROUTE
C   FBLINS = # OF LINES (ENTRIES) IN FLBASE ARRAY
C   FLBAS1 = ORIG. BASE & TIME PD FOR GIVEN FLT LEG
C   FLBAS2 = ORIG. BASE & TIME PD FOR GIVEN FLT LEG
C   FLBASE(*,?) = BASE ? (1 FOR ORIG; 2 FOR DEST) & TIME PD OF FLT
C                 LEG *
C   FLT(*) = CUM. FLT TIME OF AIRCRAFT FLYING GIVEN LEG OF A
C            MISSION
C   FLTTIM = LEG FLIGHT TIME FOR CURRENT OCCURANCE OF CURRENT
C            ROUTE
C   FLYD(*) = DESTINATION BASE OF MISSION LEG
C   FLYO(*) = ORIGIN BASE OF MISSION LEG
C   FLYTIM(*) = FLIGHT TIME BETWEEN ORIGIN AND DEST. BASES
C   GRNTIM = LEG GROUND TIME FOR CURRENT OCCURANCE OF CURRENT
C            ROUTE
C   NUMBAS = # OF BASES IN EUROPEAN THEATRE
C   NUMOD = # OF O-D PAIRS
C   OCCUR = # OF TIMES THE CURRENT ROUTE IS FLOWN IN ONE WEEK
C   OD(*,1) = ORIGIN BASE FOR ROW *
```

```
C   OD(*,2) = DESTINATION BASE FOR ROW *
C   RTBASE(*) = ICAO CODE FOR BASE * ON CURRENT ROUTE
C   RTBASES = # OF BASES ON CURRENT ROUTE
C   RTID = I.D. OF CURRENT ROUTE
C   RTSTOP(*) = STOPPING CODE FOR BASE * ON CURRENT ROUTE
C   SCHAC(*) = AIRCRAFT TYPE FOR SCHEDULE *
C   SCHDEP(*) = ORIG. DEPARTURE TIME FOR SCHEDULE *
C   SCHID(*) = ROUTE ID FOR SCHEDULE *
C
      INTEGER I,J,K,L, NUMOD, NUMBAS, RTBASES, RTID, RTSTOP(15)
      INTEGER SCHID(612), DEPCON, ARRCON, FBLINS, CAPAC
      INTEGER COUNT, CAP(500), CNT22, OCCUR
      CHARACTER*4 OD(150,2), BASE(50), RTBASE(15), SCHAC(612)
      CHARACTER*4 FLYO(560), FLYD(560), AC(10)
      CHARACTER*6 FLBASE(500,2), FLBAS1, FLBAS2
      CHARACTER*2 DEPCH, ARRCH
      REAL CUMDEM(150,7), SCHDEP(612), FLYTIM(560), DEPART(10)
      REAL DEPTIM, FLTTIM, GRNTIM, ARRTIM, MULTIP, AVGFLT
      REAL FLT(500)

      OPEN(UNIT=11,FILE='dmdeuro.dat',STATUS='OLD',ERR=91)
      OPEN(UNIT=12,FILE='baseeuro.dat',STATUS='OLD',ERR=92)
      OPEN(UNIT=13,FILE='rteeuro.dat',STATUS='OLD',ERR=93)
      OPEN(UNIT=14,FILE='schedule.raw',STATUS='OLD',ERR=94)
      OPEN(UNIT=15,FILE='fly.dat',STATUS='OLD',ERR=95)
      OPEN(UNIT=18,FILE='exampl.gms',STATUS='UNKNOWN',ERR=96)
      OPEN(UNIT=20,FILE='gams.tmp1',STATUS='UNKNOWN',ERR=98)
      OPEN(UNIT=22,FILE='gams.tmp2',STATUS='UNKNOWN',ERR=99)

      DO 10 I = 1, 150
          READ(11,801,END=91) (OD(I,J), J=1,2), (CUMDEM(I,K),
     + K=1,7)
   10 CONTINUE
   91 NUMOD = I - 1
      CLOSE(11)

       PRINT*,'NUMBER OF O-D PAIRS =',NUMOD

      WRITE(18,*) 'SET K   commodities (cargo)'
      DO 15 I = 1, NUMOD
        IF (I .EQ. 1) THEN
          WRITE(18,805) (OD(I,J), J=1,2)
        ELSE
          IF (I .EQ. NUMOD) THEN
            WRITE(18,815) (OD(I,J), J=1,2)
          ELSE
            WRITE(18,810) (OD(I,J), J=1,2)
          ENDIF
        ENDIF
   15 CONTINUE
```

```
      DO 20 I = 1, 50
          READ(12,820,END=92) BASE(I)
   20 CONTINUE
   92 NUMBAS = I - 1
      CLOSE(12)
      WRITE(18,*) ' '
      WRITE(18,*) 'SET I   airbase-time periods'


      DO 25 I = 1, NUMBAS
        IF (I .EQ. 1) THEN
          WRITE(18,824) BASE(I), BASE(I)
        ELSE
          IF (I .EQ. NUMBAS) THEN
            WRITE(18,826) BASE(I), BASE(I)
          ELSE
            WRITE(18,825) BASE(I), BASE(I)
          ENDIF
        ENDIF
   25 CONTINUE


      WRITE(18,*) ' '
      WRITE(18,*) 'ALIAS (I,IP);'
      WRITE(18,*) ' '
      WRITE(18,*) 'ALIAS (I,J);'
      WRITE(18,*) ' '
      WRITE(18,*) 'SET IK(I,K)  airbase(AB)-cargo combinations'

      DO 30 I = 1, NUMBAS
          IF (I .EQ. 1) THEN
              WRITE(18,830) BASE(I), BASE(I)
              DO 27 K = 1, NUMOD
                IF (K .EQ. 1) THEN
                  WRITE(18,809) (OD(K,J), J=1,2)
                ELSE
                  IF (K .EQ. NUMOD) THEN
                    WRITE(18,811) (OD(K,J), J=1,2)
                  ELSE
                    WRITE(18,810) (OD(K,J), J=1,2)
                  ENDIF
                ENDIF
   27         CONTINUE
            ELSE
              IF (I .EQ. NUMBAS) THEN
                  WRITE(18,840) BASE(I), BASE(I)
                  DO 28 K = 1, NUMOD
                    IF (K .EQ. 1) THEN
                      WRITE(18,809) (OD(K,J), J=1,2)
```

```
                ELSE
                  IF (K .EQ. NUMOD) THEN
                    WRITE(18,812) (OD(K,J), J=1,2)
                  ELSE
                    WRITE(18,810) (OD(K,J), J=1,2)
                  ENDIF
                ENDIF
28            CONTINUE
            ELSE
              WRITE(18,840) BASE(I), BASE(I)
              DO 29 K = 1, NUMOD
                IF (K .EQ. 1) THEN
                  WRITE(18,809) (OD(K,J), J=1,2)
                ELSE
                  IF (K .EQ. NUMOD) THEN
                    WRITE(18,811) (OD(K,J), J=1,2)
                  ELSE
                    WRITE(18,810) (OD(K,J), J=1,2)
                  ENDIF
                ENDIF
29            CONTINUE
          ENDIF
      ENDIF
30 CONTINUE

   WRITE(18,*) ' '
   WRITE(18,*) 'SET DIK(I,K)  dynamic set for IK;'
   WRITE(18,*) 'DIK(I,K) = yes;'
   WRITE(18,*) ' '
   WRITE(18,*) 'SET E1(I,J,K) arcs for cargo staying at AB'

   DO 40 I = 1, NUMBAS
     DO 35 J = 1, 7
       IF ((I .EQ. 1).AND.(J .EQ. 1)) THEN
       WRITE(18,860) BASE(I),1,BASE(I),2,
  &                  BASE(I),2,BASE(I),3,
  &                  BASE(I),3,BASE(I),4
       ELSE
         IF ((I .EQ. NUMBAS).AND.(J .EQ. 7)) THEN
         WRITE(18,880) BASE(I),19,BASE(I),20,
  &                    BASE(I),20,BASE(I),21,
  &                    BASE(I),21,BASE(I),1
         ELSE
           IF (J .EQ. 7) THEN
           WRITE(18,870) BASE(I),19,BASE(I),20,
  &                      BASE(I),20,BASE(I),21,
  &                      BASE(I),21,BASE(I),1
           ELSE
             IF (J .LE. 2) THEN
             WRITE(18,883) BASE(I),3*J-2,BASE(I),3*J-1,
```

```
     &                          BASE(I),3*J-1,BASE(I),3*J,
     &                          BASE(I),3*J,BASE(I),3*J+1
                ELSE
                  IF (J .EQ. 3) THEN
                  WRITE(18,885) BASE(I),3*J-2,BASE(I),3*J-1,
     &                          BASE(I),3*J-1,BASE(I),3*J,
     &                          BASE(I),3*J,BASE(I),3*J+1
                  ELSE
                    WRITE(18,887) BASE(I),3*J-2,BASE(I),3*J-1,
     &                            BASE(I),3*J-1,BASE(I),3*J,
     &                            BASE(I),3*J,BASE(I),3*J+1
                  ENDIF
                ENDIF
              ENDIF
            ENDIF
          ENDIF
35      CONTINUE
40    CONTINUE
      DO 45 K = 1, NUMOD
        IF (K .EQ. 1) THEN
          WRITE(18,809) (OD(K,J), J=1,2)
        ELSE
          IF (K .EQ. NUMOD) THEN
            WRITE(18,812) (OD(K,J), J=1,2)
          ELSE
            WRITE(18,810) (OD(K,J), J=1,2)
          ENDIF
        ENDIF
45    CONTINUE

      WRITE(18,*) ' '
      WRITE(18,*) 'SET Et(I,J,K)  dynamic set for E1;'
      WRITE(18,*) 'Et(I,J,K) = no;'
      WRITE(18,*) 'Et(E1) = yes;'
      WRITE(18,*) ' '
      WRITE(18,*) 'SET E2(I,J,K) arcs for A-C with cargo'

      DO 50 I = 1, 611
        READ(14,910,END=94) SCHID(I), SCHAC(I), SCHDEP(I)
50    CONTINUE
94    CLOSE(14)

      DO 60 I = 1, 560
        READ(15,920,END=95) FLYO(I), FLYD(I), FLYTIM(I)
60    CONTINUE
95    CLOSE(15)

      FBLINS = 0

      DO 90 I = 1, 49
```

```
      RTBASES = 0
      DO 70 J = 1, 15
        RTSTOP(J) = 0
        RTBASE(J) = '    '
70    CONTINUE
      READ(13,900,END=93) RTID, (RTBASE(J),RTSTOP(J), J=1,15)
      DO 80 J = 1, 15
        IF (RTSTOP(J) .GT. 0) RTBASES = RTBASES + 1
80    CONTINUE
C      PRINT*, '# OF BASES ON RTE',RTID,'  IS',RTBASES
      OCCUR = 0
      DO 82 J = 1, 611
        IF ((RTID .EQ. SCHID(J)).AND.(SCHDEP(J) .LE. 7.0)) THEN
          OCCUR = OCCUR + 1
          DEPART(OCCUR) = SCHDEP(J) * 24.
          AC(OCCUR) = SCHAC(J)
        ENDIF
82    CONTINUE
      DO 84 K = 1, OCCUR
        DEPTIM = DEPART(K)
        FLTTIM = 0.
        DO 86 J = 1, RTBASES-1
          GRNTIM = 0.
          IF (RTSTOP(J) .EQ. 6) THEN
            IF (AC(K) .EQ. 'C005') GRNTIM = 18.25
            IF (AC(K) .EQ. 'C141') GRNTIM = 17.25
            IF (AC(K) .EQ. 'C130') GRNTIM = 16.25
            IF (AC(K) .EQ. 'DC08') GRNTIM = 16.00
            IF (AC(K) .EQ. 'DC10') GRNTIM = 16.00
            IF (AC(K) .EQ. 'B747') GRNTIM = 16.00
            IF (AC(K) .EQ. 'KC10') GRNTIM = 17.25
          ELSE
            IF (RTSTOP(J) .GT. 1) THEN
              IF (AC(K) .EQ. 'C005') GRNTIM = 4.25
              IF (AC(K) .EQ. 'C141') GRNTIM = 3.25
              IF (AC(K) .EQ. 'C130') GRNTIM = 2.25
              IF (AC(K) .EQ. 'DC08') GRNTIM = 3.00
              IF (AC(K) .EQ. 'DC10') GRNTIM = 4.00
              IF (AC(K) .EQ. 'B747') GRNTIM = 4.00
              IF (AC(K) .EQ. 'KC10') GRNTIM = 3.25
            ENDIF
          ENDIF
          DEPTIM = DEPTIM + GRNTIM + FLTTIM
          IF ((RTBASE(J) .EQ. 'EXXX').OR.(RTBASE(J) .EQ.
     +         'KXXX').OR.
     &      (RTBASE(J+1) .EQ. 'EXXX').OR.(RTBASE(J+1) .EQ.
     +         'KXXX')) THEN
            FLTTIM = 0.
          ELSE
            IF (AC(K) .EQ. 'C005') MULTIP = 0.97
```

```
          IF (AC(K) .EQ. 'C141') MULTIP = 1.00
          IF (AC(K) .EQ. 'C130') MULTIP = 1.39
          IF (AC(K) .EQ. 'DC08') MULTIP = 0.93
          IF (AC(K) .EQ. 'DC10') MULTIP = 0.92
          IF (AC(K) .EQ. 'B747') MULTIP = 0.91
          IF (AC(K) .EQ. 'KC10') MULTIP = 0.92
          DO 88 L = 1, 559
            IF ((RTBASE(J).EQ.FLYO(L)).AND.
     +          (RTBASE(J+1).EQ.FLYD(L)))
     &          FLTTIM = FLYTIM(L) * MULTIP
88        CONTINUE
        ENDIF
        ARRTIM = DEPTIM + FLTTIM
        DEPCON = INT(DEPTIM/8.) + 1.
        ARRCON = INT(ARRTIM/8.) + 1.
        IF (ARRCON .LE. 21) THEN
          FBLINS = FBLINS + 1
          IF (DEPCON .EQ. 1) DEPCH = '1 '
          IF (DEPCON .EQ. 2) DEPCH = '2 '
          IF (DEPCON .EQ. 3) DEPCH = '3 '
          IF (DEPCON .EQ. 4) DEPCH = '4 '
          IF (DEPCON .EQ. 5) DEPCH = '5 '
          IF (DEPCON .EQ. 6) DEPCH = '6 '
          IF (DEPCON .EQ. 7) DEPCH = '7 '
          IF (DEPCON .EQ. 8) DEPCH = '8 '
          IF (DEPCON .EQ. 9) DEPCH = '9 '
          IF (DEPCON .EQ. 10) DEPCH = '10'
          IF (DEPCON .EQ. 11) DEPCH = '11'
          IF (DEPCON .EQ. 12) DEPCH = '12'
          IF (DEPCON .EQ. 13) DEPCH = '13'
          IF (DEPCON .EQ. 14) DEPCH = '14'
          IF (DEPCON .EQ. 15) DEPCH = '15'
          IF (DEPCON .EQ. 16) DEPCH = '16'
          IF (DEPCON .EQ. 17) DEPCH = '17'
          IF (DEPCON .EQ. 18) DEPCH = '18'
          IF (DEPCON .EQ. 19) DEPCH = '19'
          IF (DEPCON .EQ. 20) DEPCH = '20'
          IF (DEPCON .EQ. 21) DEPCH = '21'
          IF (ARRCON .EQ. 1) ARRCH = '1 '
          IF (ARRCON .EQ. 2) ARRCH = '2 '
          IF (ARRCON .EQ. 3) ARRCH = '3 '
          IF (ARRCON .EQ. 4) ARRCH = '4 '
          IF (ARRCON .EQ. 5) ARRCH = '5 '
          IF (ARRCON .EQ. 6) ARRCH = '6 '
          IF (ARRCON .EQ. 7) ARRCH = '7 '
          IF (ARRCON .EQ. 8) ARRCH = '8 '
          IF (ARRCON .EQ. 9) ARRCH = '9 '
          IF (ARRCON .EQ. 10) ARRCH = '10'
          IF (ARRCON .EQ. 11) ARRCH = '11'
          IF (ARRCON .EQ. 12) ARRCH = '12'
```

```
            IF (ARRCON .EQ. 13) ARRCH = '13'
            IF (ARRCON .EQ. 14) ARRCH = '14'
            IF (ARRCON .EQ. 15) ARRCH = '15'
            IF (ARRCON .EQ. 16) ARRCH = '16'
            IF (ARRCON .EQ. 17) ARRCH = '17'
            IF (ARRCON .EQ. 18) ARRCH = '18'
            IF (ARRCON .EQ. 19) ARRCH = '19'
            IF (ARRCON .EQ. 20) ARRCH = '20'
            IF (ARRCON .EQ. 21) ARRCH = '21'
            FLBASE(FBLINS,1) = RTBASE(J) // DEPCH
            FLBASE(FBLINS,2) = RTBASE(J+1) // ARRCH

            FLT(FBLINS) = FLTTIM
            IF (AC(K) .EQ. 'C005') CAP(FBLINS) = 50
            IF (AC(K) .EQ. 'C141') CAP(FBLINS) = 18
            IF (AC(K) .EQ. 'C130') CAP(FBLINS) = 7
            IF (AC(K) .EQ. 'DC08') CAP(FBLINS) = 25
            IF (AC(K) .EQ. 'DC10') CAP(FBLINS) = 40
            IF (AC(K) .EQ. 'B747') CAP(FBLINS) = 71
            IF (AC(K) .EQ. 'KC10') CAP(FBLINS) = 30
          ENDIF
 86     CONTINUE
 84   CONTINUE

 90 CONTINUE
 93 CONTINUE

    CNT22 = 0
    DO 104 J = 1, FBLINS
      COUNT = 1
      IF (J .EQ. 1) GO TO 102
      DO 101 I = 1, J-1
          IF ((FLBASE(I,1) .EQ. FLBASE(J,1)).AND.
   &            (FLBASE(I,2) .EQ. FLBASE(J,2))) GO TO 104
101   CONTINUE
102   DO 103 I = J+1, FBLINS
        IF ((FLBASE(I,1) .EQ. FLBASE(J,1)).AND.
   &            (FLBASE(I,2) .EQ. FLBASE(J,2))) THEN
            COUNT = COUNT + 1
            FLT(J) = FLT(J) + FLT(I)
            CAP(J) = CAP(J) + CAP(I)
        ENDIF
103   CONTINUE
      CNT22 = CNT22 + 1
      AVGFLT = FLT(J)/COUNT
      WRITE(22,960) FLBASE(J,1), FLBASE(J,2), AVGFLT, CAP(J)
104 CONTINUE

    REWIND 22
    DO 106 J = 1, CNT22
```

```
      READ(22,960) FLBAS1, FLBAS2, AVGFLT, CAPAC
      IF (J .EQ. 1) THEN
          WRITE(18,925) FLBAS1, FLBAS2
        ELSE
          IF (J .EQ. CNT22) THEN
              WRITE(18,929) FLBAS1, FLBAS2
            ELSE
              WRITE(18,927) FLBAS1, FLBAS2
          ENDIF
      ENDIF
106 CONTINUE

    DO 108 K = 1, NUMOD
      IF (K .EQ. 1) THEN
        WRITE(18,809) (OD(K,J), J=1,2)
      ELSE
        IF (K .EQ. NUMOD) THEN
          WRITE(18,812) (OD(K,J), J=1,2)
        ELSE
          WRITE(18,810) (OD(K,J), J=1,2)
        ENDIF
      ENDIF
108 CONTINUE

    CLOSE(13)

    WRITE(18,*) ' '
    WRITE(18,*) ' SET Es(I,J,K)  dynamic set for E2;'
    WRITE(18,*) ' Es(I,J,K) = no;'
    WRITE(18,*) ' Es(E2) = yes;'
    WRITE(18,*) ' '
    WRITE(18,*) ' SET E(I,J,K) set of all arcs (Et and Es);'
    WRITE(18,*) ' E(I,J,K) = Et(I,J,K) + Es(I,J,K);'
    WRITE(18,*) ' '
    WRITE(18,*) ' SET E3(I,J) arcs representing aircraft'

    REWIND 22
    DO 110 J = 1, CNT22
      READ(22,960) FLBAS1, FLBAS2, AVGFLT, CAPAC
      IF (J .EQ. 1) THEN
          WRITE(18,935) FLBAS1, FLBAS2
        ELSE
          IF (J .EQ. CNT22) THEN
              WRITE(18,939) FLBAS1, FLBAS2
            ELSE
              WRITE(18,927) FLBAS1, FLBAS2
          ENDIF
      ENDIF
110 CONTINUE
```

```
      WRITE(18,*) ' '
      WRITE(18,*)'SET SIKN(I,K) airbase supply nodes'

      DO 120 I = 1, NUMOD
        WRITE(20,1000) (OD(I,1), OD(I,1), OD(I,2), J=1, 7)
        IF (I .EQ. 1) THEN
          WRITE(18,1002) (OD(I,1), OD(I,1), OD(I,2), J=1, 7)
        ELSE
          IF (I .EQ. NUMOD) THEN
            WRITE(18,1006) (OD(I,1), OD(I,1), OD(I,2), J=1, 7)
          ELSE
            WRITE(18,1004) (OD(I,1), OD(I,1), OD(I,2), J=1, 7)
          ENDIF
        ENDIF
  120 CONTINUE
   98 CONTINUE

      WRITE(18,*) ' '
      WRITE(18,*) 'SET SUPNODE(I,K)  dynamic set for SIKN;'
      WRITE(18,*) 'SUPNODE(I,K) = no;'
      WRITE(18,*) 'SUPNODE(SIKN) = yes;'
      WRITE(18,*) ' '
      WRITE(18,*) 'SET DIKN(I,K) airbase demand nodes'

      DO 130 I = 1, NUMOD
        IF (I .EQ. 1) THEN
          WRITE(18,1012) OD(I,2), OD(I,2), OD(I,1), OD(I,2)
        ELSE
          IF (I .EQ. NUMOD) THEN
          WRITE(18,1016) OD(I,2), OD(I,2), OD(I,1), OD(I,2)
          ELSE
          WRITE(18,1014) OD(I,2), OD(I,2), OD(I,1), OD(I,2)
          ENDIF
        ENDIF
  130 CONTINUE

      WRITE(18,*) ' '
      WRITE(18,*) 'SET DMDNODE(I,K)  dynamic set for DIKN;'
      WRITE(18,*) 'DMDNODE(I,K) = no;'
      WRITE(18,*) 'DMDNODE(DIKN) = yes;'
      WRITE(18,*) ' '
      WRITE(18,*) 'SET ZIKN(I,K) neither dmd nor sup nodes;'
      WRITE(18,*) 'ZIKN(I,K) = DIK(I,K) - SUPNODE(I,K) -
     + DMDNODE(I,K);'
      WRITE(18,*) ' '
      WRITE(18,*) 'PARAMETER C(I,J,K)  delay;'
      WRITE(18,*) ' '
      WRITE(18,*) 'C(I,J,K) = 0;'
      WRITE(18,*) ' '
      WRITE(18,*) 'C(I,J,K)$Et(I,J,K) = 8;'
```

```
      WRITE(18,*) ' '

CCC      WRITE(18,*) 'C(I,J,K)$Es(I,J,K) = (flt time)'
      REWIND 22
      DO 140 J = 1, CNT22
        READ(22,960) FLBAS1, FLBAS2, AVGFLT, CAPAC
        WRITE(18,1020) FLBAS1, FLBAS2, AVGFLT
  140 CONTINUE


      WRITE(18,*) ' '
      WRITE(18,*) 'PARAMETER S(I,K) the supply at node SIKN'

      REWIND 20

      DO 150 I = 1, NUMOD
        READ(20,1000) (OD(I,1), OD(I,1), OD(I,2), J=1, 7)
        IF (I .EQ. 1) THEN
          WRITE(18,1028) OD(I,1), OD(I,1), OD(I,2), CUMDEM(I,1),
     &                (OD(I,1), OD(I,1), OD(I,2),
     &                 CUMDEM(I,J)-CUMDEM(I,J-1), J=2, 7)
        ELSE
          IF (I .EQ. NUMOD) THEN
            WRITE(18,1032) OD(I,1), OD(I,1), OD(I,2),
     +           CUMDEM(I,1),
     &                (OD(I,1), OD(I,1), OD(I,2),
     &                 CUMDEM(I,J)-CUMDEM(I,J-1), J=2, 7)
          ELSE
            WRITE(18,1030) OD(I,1), OD(I,1), OD(I,2),
     +           CUMDEM(I,1),
     &                (OD(I,1), OD(I,1), OD(I,2),
     &                 CUMDEM(I,J)-CUMDEM(I,J-1), J=2, 7)
          ENDIF
        ENDIF
  150 CONTINUE
      CLOSE(20)

      WRITE(18,*) ' '
      WRITE(18,*) 'PARAMETER CAP(I,J)  aircraft capacity'

      REWIND 22
      DO 160 J = 1, CNT22
        READ(22,960) FLBAS1, FLBAS2, AVGFLT, CAPAC
        IF (J .EQ. 1) THEN
            WRITE(18,945) FLBAS1, FLBAS2, CAPAC
          ELSE
            IF (J .EQ. CNT22) THEN
                WRITE(18,949) FLBAS1, FLBAS2, CAPAC
              ELSE
```

```
              WRITE(18,947) FLBAS1, FLBAS2, CAPAC
          ENDIF
        ENDIF
160 CONTINUE
 99 CLOSE(22)

    WRITE(18,*) ' '
    WRITE(18,*) 'VARIABLE'
    WRITE(18,*) 'Z  total delay'
    WRITE(18,*) ' '
    WRITE(18,*) 'POSITIVE VARIABLES'
    WRITE(18,*) 'X(I,J,K) shipment quantity'
    WRITE(18,*) 'SUP(K)  total supply for each cargo K'
    WRITE(18,*) 'DEL(K)  amount delivered for each cargo'
    WRITE(18,*) 'UNDEL(K) amount not delivered for each cargo'
    WRITE(18,*) ' '
    WRITE(18,*) 'EQUATIONS'
    WRITE(18,*) 'DELAY  objective function'
    WRITE(18,*) 'SUMS(K)  total supply for each cargo K'
    WRITE(18,*) 'SUPLY(IP,K)  conserv. of flow for sup. nodes'
    WRITE(18,*) 'DEMND(IP,K)  conserv. of flow for dmd. nodes'
    WRITE(18,*) 'DELIVER(K)   amount delivered for each cargo'
    WRITE(18,*) 'UNDELIVER(K) amount not delivered'
    WRITE(18,*) 'BAL(IP,K)  conserv. of flow for ZIKN nodes'
    WRITE(18,*) 'UB(I,J)  upper bound capac. for aircraft;'
    WRITE(18,*) ' '
    WRITE(18,*) 'DELAY .. Z =E= SUM((I,J,K)$E(I,J,K),'
    WRITE(18,*) '               C(I,J,K)*X(I,J,K));'
    WRITE(18,*) ' '
    WRITE(18,*) 'SUMS(K) .. SUP(K) =E= SUM(I,S(I,K));'
    WRITE(18,*) ' '
    WRITE(18,*) 'SUPLY(IP,K)$SIKN(IP,K)..'
    WRITE(18,*) '           SUM(J,X(IP,J,K)$E(IP,J,K)) -'
    WRITE(18,*) '           SUM(I,X(I,IP,K)$E(I,IP,K))'
    WRITE(18,*) '           =E= S(IP,K);'
    WRITE(18,*) ' '
    WRITE(18,*) 'DEMND(IP,K)$DIKN(IP,K)..'
    WRITE(18,*) '           SUM(J,X(IP,J,K)$E(IP,J,K)) -'
    WRITE(18,*) 'SUM(I,X(I,IP,K)$E(I,IP,K))'
    WRITE(18,*) '           =G= -SUP(K);'
    WRITE(18,*) ' '
    WRITE(18,*) 'DELIVER(K) .. DEL(K) =E= SUM((I,IP)$E3(I,IP),'
    WRITE(18,*) 'X(I,IP,K)$DIKN(IP,K));'
    WRITE(18,*) ' '
    WRITE(18,*) 'UNDELIVER(K) .. UNDEL(K) =E= SUP(K) - DEL(K);'
    WRITE(18,*) ' '
    WRITE(18,*) 'BAL(IP,K)$ZIKN(IP,K) ..'
    WRITE(18,*) '          SUM(J,X(IP,J,K)$E(IP,J,K)) -'
    WRITE(18,*) 'SUM(I,X(I,IP,K)$E(I,IP,K))'
    WRITE(18,*) '          =E= 0;'
```

```
      WRITE(18,*) ' '
      WRITE(18,*) 'UB( E3(I,J) ) ..  SUM(K, X(I,J,K))'
      WRITE(18,*) '          =L= CAP(E3);'
      WRITE(18,*) ' '
      WRITE(18,*) 'MODEL MMCF /ALL/;'
      WRITE(18,*) ' '
      WRITE(18,*) 'OPTION ITERLIM = 10000, RESLIM = 100000; '
      WRITE(18,*) 'OPTION LIMROW = 0, LIMCOL = 0;'
      WRITE(18,*) ' '
      WRITE(18,*) 'SOLVE MMCF USING LP MINIMIZING Z;'
   96 CLOSE(18)

C*******************format statements:
  801 FORMAT(A4,1X,A4,7(1X,F6.2))
  805 FORMAT(1X, '/', A4,A4,', ')
  809 FORMAT(1X, '(', A4,A4,', ')
  810 FORMAT(1X, ' ', A4,A4,', ')
  811 FORMAT(1X, ' ', A4,A4,'),')
  812 FORMAT(1X, ' ', A4,A4,')/;')
  815 FORMAT(1X, ' ', A4,A4,'/;')
  820 FORMAT(4X, A4)
  824 FORMAT(1X, '/', A4, '1 * ', A4, '21, ')
  825 FORMAT(1X, ' ', A4, '1 * ', A4, '21, ')
  826 FORMAT(1X, ' ', A4, '1 * ', A4, '21/;')
  830 FORMAT(1X,'/(',A4,'1 * ',A4,'21).')
  840 FORMAT(1X,' (',A4,'1 * ',A4,'21).')
  860 FORMAT(1X, '/(', 3(A4, I1, '.', A4, I1, ', '))
  870 FORMAT(1X,2(A4,I2, '.',A4,I2,', '), A4,I2,'.',A4,I1,',')
  880 FORMAT(1X,2(A4,I2, '.',A4,I2,', '), A4,I2,'.',A4,I1,').')
  883 FORMAT(1X, 3(A4, I1, '.', A4, I1, ', '))
  885 FORMAT(1X,2(A4,I1, '.',A4,I1,', '), A4,I1,'.',A4,I2,',')
  887 FORMAT(1X, 3(A4, I2, '.', A4, I2, ', '))
  900 FORMAT(I3, 15(1X,A4,I1))
  910 FORMAT(I3,2X,A4,2X,F4.1)
  920 FORMAT(2(A4,1X),6X,F4.1)
  925 FORMAT(1X, '/(', A6, '.', A6, ', ')
  927 FORMAT(1X, '  ', A6, '.', A6, ', ')
  929 FORMAT(1X, '  ', A6, '.', A6, ').')
  935 FORMAT(1X, '/', A6, '.', A6, ', ')
  939 FORMAT(1X, '  ', A6, '.', A6, '/;')
  945 FORMAT(1X, '/', A6, '.', A6, ' ', I3, ', ')
  947 FORMAT(1X, ' ', A6, '.', A6, ' ', I3, ', ')
  949 FORMAT(1X, ' ', A6, '.', A6, ' ', I3, '/;')
  960 FORMAT(1X, A6, A6, F6.2, 1X, I3)
 1000 FORMAT(1X,A4,'1.',2A4,/,1X,A4,'4.',2A4,/,1X,A4,'7.',
     +2A4,/,1X, A4,'10.',2A4,/,1X,A4,'13.',2A4,/,1X,A4,'16.',
     +2A4,/,1X,A4,'19.',2A4)
 1002 FORMAT(1X,'/',A4,'1.',2A4,', ',A4,'4.',2A4,', ',A4,
     +'7.',2A4,', ',/,1X,A4,'10.',2A4,', ',A4,'13.',2A4,',
     +',A4,'16.',2A4,', ',/,1X,A4,'19.',2A4,', ')
```

```
1004 FORMAT(1X,A4,'1.',2A4,', ',A4,'4.',2A4,', ',A4,'7.',
    +2A4,', ',/,1X,A4,'10.',2A4,', ',A4,'13.',2A4,', ',A4,
    +'16.',2A4,', ',/,1X,A4,'19.',2A4,', ')
1006 FORMAT(1X,A4,'1.',2A4,', ',A4,'4.',2A4,', ',A4,'7.',
    +2A4,', ',/,1X,A4,'10.',2A4,', ',A4,'13.',2A4,', ',A4,
    +'16.',2A4,', ',/,1X,A4,'19.',2A4,'/;')
1012 FORMAT(1X,'/(',A4,'1 * ',A4,'21).',2A4,',')
1014 FORMAT(1X,'(',A4,'1 * ',A4,'21).',2A4,',')
1016 FORMAT(1X,'(',A4,'1 * ',A4,'21).',2A4,'/;')
1020 FORMAT(1X,'C("',A6,'","',A6,'",K)=',F5.1,';')
1028 FORMAT(X,'/',A4,'1.',2A4,X,F6.2,', ',A4,'4.',2A4,X,
    +F6.2,', ',A4,'7.',2A4,X,F6.2,', ',/,X,A4,
    +'10.',2A4,X,F6.2,', ',A4,'13.',2A4,X,
    & F6.2,', ',A4,'16.',2A4,X,F6.2,', ',/,X,A4,
    +'19.',2A4,X,F6.2,', ')
1030 FORMAT(X,A4,'1.',2A4,X,F6.2,', ',A4,'4.',2A4,X,
    +F6.2,', ',A4,'7.',2A4,X,F6.2,', ',/,X,A4,'10.',2A4,
    +X,F6.2,', ',A4,'13.',2A4,X,F6.2,', ',A4,'16.',2A4,X,
    +F6.2,', ',/,X,A4,'19.',2A4,X,F6.2,', ')
1032 FORMAT(X,A4,'1.',2A4,X,F6.2,', ',A4,'4.',2A4,X,
    +F6.2,', ',A4,'7.',2A4,X,F6.2,', ',/,X,A4,'10.',2A4,
    +X,F6.2,', ',A4,'13.',2A4,X,F6.2, ', ',A4,'16.',2A4,
    +X,F6.2,', ',/,X,A4,'19.',2A4,X,F6.2,'/;')
C*****
      STOP
      END
```

Appendix K: GAMS Program

This appendix shows an extract of the GAMS program used for the subproblem in this research. This GAMS program is created by the FORTRAN program "GAMS.FOR" (shown in Appendix J).

```
SET K   commodities
 /EDARKNGU,
  EDARLGIR,
  EDARLICZ,
  EDARLIRN,
  EDAROEDR,
  EGUNKNGU,
  EGUNLTAG,
  KCHSEDAF,
  KDOVLGIR,
  KDOVLIPA,
  KDOVOEDR,
  KNGULIPA,
  KTIKLGIR,
  KTIKLIPA,
  KTIKLTAG,
  KTIKOEDR,
  KTIKOERY,
  LETOKDOV,
  LETOKTIK,
  LETOKWRI/;

SET I   airbase-time periods
 /BIKF1 * BIKF21,
  CYQX1 * CYQX21,
  EDAF1 * EDAF21,
  EDAR1 * EDAR21,
  EGUN1 * EGUN21,
  EXXX1 * EXXX21,
  FTTJ1 * FTTJ21,
  FZAA1 * FZAA21,
  GLRB1 * GLRB21,
  GOOY1 * GOOY21,
  HKNA1 * HKNA21,
  HSSS1 * HSSS21,
  KCHS1 * KCHS21,
  KDOV1 * KDOV21,
  KNGU1 * KNGU21,
  KSUU1 * KSUU21,
  KTIK1 * KTIK21,
  KWRI1 * KWRI21,
```

```
  KXXX1 * KXXX21,
  LCRA1 * LCRA21,
  LERT1 * LERT21,
  LETO1 * LETO21,
  LGIR1 * LGIR21,
  LICZ1 * LICZ21,
  LIPA1 * LIPA21,
  LIRN1 * LIRN21,
  LIRP1 * LIRP21,
  LLBG1 * LLBG21,
  LPLA1 * LPLA21,
  LTAG1 * LTAG21,
  OBBI1 * OBBI21,
  OEDR1 * OEDR21,
  OERY1 * OERY21,
  OJAF1 * OJAF21,
  OKBK1 * OKBK21,
  OMFJ1 * OMFJ21/;

ALIAS (I,IP);

ALIAS (I,J);

SET IK(I,K)  airbase-commodity combinations
 /(BIKF1 * BIKF21).
 (EDARKNGU,
  EDARLGIR,
  EDARLICZ,
  EDARLIRN,
  EDAROEDR,
  EGUNKNGU,
  EGUNLTAG,
  KCHSEDAF,
  KDOVLGIR,
  KDOVLIPA,
  KDOVOEDR,
  KNGULIPA,
  KTIKLGIR,
  KTIKLIPA,
  KTIKLTAG,
  KTIKOEDR,
  KTIKOERY,
  LETOKDOV,
  LETOKTIK,
  LETOKWRI),
  (CYQX1 * CYQX21).
 (EDARKNGU,
  EDARLGIR,
  EDARLICZ,
  EDARLIRN,
```

```
 EDAROEDR,
 EGUNKNGU,
 EGUNLTAG,
 KCHSEDAF,
 KDOVLGIR,
 KDOVLIPA,
 KDOVOEDR,
 KNGULIPA,
 KTIKLGIR,
 KTIKLIPA,
 KTIKLTAG,
 KTIKOEDR,
 KTIKOERY,
 LETOKDOV,
 LETOKTIK,
 LETOKWRI),
 (EDAF1 * EDAF21).
(EDARKNGU,
 EDARLGIR,
 EDARLICZ,
 EDARLIRN,
 EDAROEDR,
 EGUNKNGU,
 EGUNLTAG,
 KCHSEDAF,
 KDOVLGIR,
 KDOVLIPA,
 KDOVOEDR,
 KNGULIPA,
 KTIKLGIR,
 KTIKLIPA,
 KTIKLTAG,
 KTIKOEDR,
 KTIKOERY,
 LETOKDOV,
 LETOKTIK,
 LETOKWRI),
     .
     .
     .
     .
     .

 (OKBK1 * OKBK21).
(EDARKNGU,
 EDARLGIR,
 EDARLICZ,
 EDARLIRN,
 EDAROEDR,
 EGUNKNGU,
```

```
  EGUNLTAG,
  KCHSEDAF,
  KDOVLGIR,
  KDOVLIPA,
  KDOVOEDR,
  KNGULIPA,
  KTIKLGIR,
  KTIKLIPA,
  KTIKLTAG,
  KTIKOEDR,
  KTIKOERY,
  LETOKDOV,
  LETOKTIK,
  LETOKWRI),
  (OMFJ1 * OMFJ21).
 (EDARKNGU,
  EDARLGIR,
  EDARLICZ,
  EDARLIRN,
  EDAROEDR,
  EGUNKNGU,
  EGUNLTAG,
  KCHSEDAF,
  KDOVLGIR,
  KDOVLIPA,
  KDOVOEDR,
  KNGULIPA,
  KTIKLGIR,
  KTIKLIPA,
  KTIKLTAG,
  KTIKOEDR,
  KTIKOERY,
  LETOKDOV,
  LETOKTIK,
  LETOKWRI)/;

SET DIK(I,K)  dynamic set for IK;
DIK(I,K) = yes;

SET E1(I,J,K) arcs for commods staying at an airbase
 /(BIKF1.BIKF2, BIKF2.BIKF3, BIKF3.BIKF4,
 BIKF4.BIKF5, BIKF5.BIKF6, BIKF6.BIKF7,
 BIKF7.BIKF8, BIKF8.BIKF9, BIKF9.BIKF10,
 BIKF10.BIKF11, BIKF11.BIKF12, BIKF12.BIKF13,
 BIKF13.BIKF14, BIKF14.BIKF15, BIKF15.BIKF16,
 BIKF16.BIKF17, BIKF17.BIKF18, BIKF18.BIKF19,
 BIKF19.BIKF20, BIKF20.BIKF21, BIKF21.BIKF1,
 CYQX1.CYQX2, CYQX2.CYQX3, CYQX3.CYQX4,
 CYQX4.CYQX5, CYQX5.CYQX6, CYQX6.CYQX7,
 CYQX7.CYQX8, CYQX8.CYQX9, CYQX9.CYQX10,
```

```
 CYQX10.CYQX11, CYQX11.CYQX12, CYQX12.CYQX13,
 CYQX13.CYQX14, CYQX14.CYQX15, CYQX15.CYQX16,
 CYQX16.CYQX17, CYQX17.CYQX18, CYQX18.CYQX19,
 CYQX19.CYQX20, CYQX20.CYQX21, CYQX21.CYQX1,
      .
      .
      .
      .
      .
 OKBK10.OKBK11, OKBK11.OKBK12, OKBK12.OKBK13,
 OKBK13.OKBK14, OKBK14.OKBK15, OKBK15.OKBK16,
 OKBK16.OKBK17, OKBK17.OKBK18, OKBK18.OKBK19,
 OKBK19.OKBK20, OKBK20.OKBK21, OKBK21.OKBK1,
 OMFJ1.OMFJ2, OMFJ2.OMFJ3, OMFJ3.OMFJ4,
 OMFJ4.OMFJ5, OMFJ5.OMFJ6, OMFJ6.OMFJ7,
 OMFJ7.OMFJ8, OMFJ8.OMFJ9, OMFJ9.OMFJ10,
 OMFJ10.OMFJ11, OMFJ11.OMFJ12, OMFJ12.OMFJ13,
 OMFJ13.OMFJ14, OMFJ14.OMFJ15, OMFJ15.OMFJ16,
 OMFJ16.OMFJ17, OMFJ17.OMFJ18, OMFJ18.OMFJ19,
 OMFJ19.OMFJ20, OMFJ20.OMFJ21, OMFJ21.OMFJ1).
 (EDARKNGU,
  EDARLGIR,
  EDARLICZ,
  EDARLIRN,
  EDAROEDR,
  EGUNKNGU,
  EGUNLTAG,
  KCHSEDAF,
  KDOVLGIR,
  KDOVLIPA,
  KDOVOEDR,
  KNGULIPA,
  KTIKLGIR,
  KTIKLIPA,
  KTIKLTAG,
  KTIKOEDR,
  KTIKOERY,
  LETOKDOV,
  LETOKTIK,
  LETOKWRI)/;

SET Et(I,J,K)  dynamic set for E1;
Et(I,J,K) = no;
Et(E1) = yes;

SET E2(I,J,K) arcs representing A-C with commodits
 /(EXXX10.KTIK10,
   KTIK11.CYQX11,
   CYQX12.EDAR13,
```

```
   EDAR13.EXXX13,
   KSUU11.KTIK11,
   KTIK12.KDOV12,
   KDOV14.EDAF15,
   EDAF18.KDOV19,
   KDOV21.KTIK21,
   KSUU14.KTIK14,
      .
      .
      .
      .
      .

   EDAR2 .EDAF2 ,
   EDAF10.EDAR10,
   EDAR10.EDAF10,
   EDAF17.EDAR17,
   EDAR17.EDAF17,
   KDOV1 .EDAR1 ,
   EDAR2 .LLBG3 ,
   LLBG3 .EDAR4 ,
   EDAR4 .KDOV5 ,
   KNGU20.LETO21).
  (EDARKNGU,
   EDARLGIR,
   EDARLICZ,
   EDARLIRN,
   EDAROEDR,
   EGUNKNGU,
   EGUNLTAG,
   KCHSEDAF,
   KDOVLGIR,
   KDOVLIPA,
   KDOVOEDR,
   KNGULIPA,
   KTIKLGIR,
   KTIKLIPA,
   KTIKLTAG,
   KTIKOEDR,
   KTIKOERY,
   LETOKDOV,
   LETOKTIK,
   LETOKWRI)/;

SET Es(I,J,K)  dynamic set for E2;
Es(I,J,K) = no;
Es(E2) = yes;

SET E(I,J,K) set of all arcs (union of Et and Es);
E(I,J,K) = Et(I,J,K) + Es(I,J,K);
```

```
 SET E3(I,J)  arcs representing aircraft
 /EXXX10.KTIK10,
   KTIK11.CYQX11,
   CYQX12.EDAR13,
   EDAR13.EXXX13,
   KSUU11.KTIK11,
   KTIK12.KDOV12,
   KDOV14.EDAF15,
   EDAF18.KDOV19,
      .
      .
      .
      .
      .
   EDAR2 .EDAF2 ,
   EDAF10.EDAR10,
   EDAR10.EDAF10,
   EDAF17.EDAR17,
   EDAR17.EDAF17,
   KDOV1 .EDAR1 ,
   EDAR2 .LLBG3 ,
   LLBG3 .EDAR4 ,
   EDAR4 .KDOV5 ,
   KNGU20.LETO21/;

SET SIKN(I,K) airbase supply nodes for all commoditys
 /EDAR1.EDARKNGU, EDAR4.EDARKNGU, EDAR7.EDARKNGU,
 EDAR10.EDARKNGU, EDAR13.EDARKNGU, EDAR16.EDARKNGU,
 EDAR19.EDARKNGU,
 EDAR1.EDARLGIR, EDAR4.EDARLGIR, EDAR7.EDARLGIR,
 EDAR10.EDARLGIR, EDAR13.EDARLGIR, EDAR16.EDARLGIR,
 EDAR19.EDARLGIR,
 EDAR1.EDARLICZ, EDAR4.EDARLICZ, EDAR7.EDARLICZ,
 EDAR10.EDARLICZ, EDAR13.EDARLICZ, EDAR16.EDARLICZ,
 EDAR19.EDARLICZ,
 EDAR1.EDARLIRN, EDAR4.EDARLIRN, EDAR7.EDARLIRN,
 EDAR10.EDARLIRN, EDAR13.EDARLIRN, EDAR16.EDARLIRN,
 EDAR19.EDARLIRN,
      .
      .
      .
      .
      .
 LETO1.LETOKDOV, LETO4.LETOKDOV, LETO7.LETOKDOV,
 LETO10.LETOKDOV, LETO13.LETOKDOV, LETO16.LETOKDOV,
 LETO19.LETOKDOV,
 LETO1.LETOKTIK, LETO4.LETOKTIK, LETO7.LETOKTIK,
 LETO10.LETOKTIK, LETO13.LETOKTIK, LETO16.LETOKTIK,
```

```
 LETO19.LETOKTIK,
 LETO1.LETOKWRI, LETO4.LETOKWRI, LETO7.LETOKWRI,
 LETO10.LETOKWRI, LETO13.LETOKWRI, LETO16.LETOKWRI,
 LETO19.LETOKWRI/;

SET SUPNODE(I,K)  dynamic set for SIKN;
SUPNODE(I,K) = no;
SUPNODE(SIKN) = yes;

SET DIKN(I,K) airbase demand nodes for all commodits
 /(KNGU1 * KNGU21).EDARKNGU,
 (LGIR1 * LGIR21).EDARLGIR,
 (LICZ1 * LICZ21).EDARLICZ,
 (LIRN1 * LIRN21).EDARLIRN,
 (OEDR1 * OEDR21).EDAROEDR,
 (KNGU1 * KNGU21).EGUNKNGU,
 (LTAG1 * LTAG21).EGUNLTAG,
 (EDAF1 * EDAF21).KCHSEDAF,
 (LGIR1 * LGIR21).KDOVLGIR,
 (LIPA1 * LIPA21).KDOVLIPA,
 (OEDR1 * OEDR21).KDOVOEDR,
 (LIPA1 * LIPA21).KNGULIPA,
 (LGIR1 * LGIR21).KTIKLGIR,
 (LIPA1 * LIPA21).KTIKLIPA,
 (LTAG1 * LTAG21).KTIKLTAG,
 (OEDR1 * OEDR21).KTIKOEDR,
 (OERY1 * OERY21).KTIKOERY,
 (KDOV1 * KDOV21).LETOKDOV,
 (KTIK1 * KTIK21).LETOKTIK,
 (KWRI1 * KWRI21).LETOKWRI/;

SET DMDNODE(I,K)  dynamic set for DIKN;
DMDNODE(I,K) = no;
DMDNODE(DIKN) = yes;

SET ZIKN(I,K)  neither demand nor supply nodes;
ZIKN(I,K) = DIK(I,K) - SUPNODE(I,K) - DMDNODE(I,K);

PARAMETER C(I,J,K)  delay;

C(I,J,K) = 0;

C(I,J,K)$Et(I,J,K) = 8;

 C("EXXX10","KTIK10",K)=  0.0;
 C("KTIK11","CYQX11",K)=  4.7;
 C("CYQX12","EDAR13",K)=  6.1;
 C("EDAR13","EXXX13",K)=  0.0;
 C("KSUU11","KTIK11",K)=  2.9;
 C("KTIK12","KDOV12",K)=  2.8;
```

```
C("KDOV14","EDAF15",K)=  7.7;
C("EDAF18","KDOV19",K)=  9.6;
C("KDOV21","KTIK21",K)=  3.1;
C("KSUU14","KTIK14",K)=  2.9;
     .
     .
     .
     .
     .

C("EDAF10","EDAR10",K)=  0.1;
C("EDAR10","EDAF10",K)=  0.1;
C("EDAF17","EDAR17",K)=  0.1;
C("EDAR17","EDAF17",K)=  0.1;
C("KDOV1 ","EDAR1 ",K)=  7.9;
C("EDAR2 ","LLBG3 ",K)=  4.2;
C("LLBG3 ","EDAR4 ",K)=  5.2;
C("EDAR4 ","KDOV5 ",K)=  9.2;
C("KNGU20","LETO21",K)=  8.1;

PARAMETER S(I,K) the supply at node SIKN
 /EDAR1.EDARKNGU   0.24, EDAR4.EDARKNGU   0.24,
EDAR7.EDARKNGU   0.24,
 EDAR10.EDARKNGU   0.24, EDAR13.EDARKNGU   0.24,
EDAR16.EDARKNGU   0.24,
 EDAR19.EDARKNGU   0.24,
 EDAR1.EDARLGIR   0.30, EDAR4.EDARLGIR   0.29,
EDAR7.EDARLGIR   0.30,
 EDAR10.EDARLGIR   0.30, EDAR13.EDARLGIR   0.29,
EDAR16.EDARLGIR   0.30,
 EDAR19.EDARLGIR   0.30,
 EDAR1.EDARLICZ   0.18, EDAR4.EDARLICZ   0.18,
EDAR7.EDARLICZ   0.18,
 EDAR10.EDARLICZ   0.18, EDAR13.EDARLICZ   0.18,
EDAR16.EDARLICZ   0.18,
 EDAR19.EDARLICZ   0.18,
 EDAR1.EDARLIRN   0.18, EDAR4.EDARLIRN   0.19,
EDAR7.EDARLIRN   0.18,
 EDAR10.EDARLIRN   0.18, EDAR13.EDARLIRN   0.19,
EDAR16.EDARLIRN   0.18,
 EDAR19.EDARLIRN   0.18,
     .
     .
     .
     .
     .

 LETO1.LETOKDOV   8.19, LETO4.LETOKDOV   8.18,
LETO7.LETOKDOV   8.19,
```

```
 LETO10.LETOKDOV   8.19, LETO13.LETOKDOV   8.18,
LETO16.LETOKDOV   8.19,
 LETO19.LETOKDOV   8.19,
 LETO1.LETOKTIK   0.77, LETO4.LETOKTIK   0.77,
LETO7.LETOKTIK   0.77,
 LETO10.LETOKTIK   0.77, LETO13.LETOKTIK   0.77,
LETO16.LETOKTIK   0.77,
 LETO19.LETOKTIK   0.77,
 LETO1.LETOKWRI   1.16, LETO4.LETOKWRI   1.16,
LETO7.LETOKWRI   1.16,
 LETO10.LETOKWRI   1.16, LETO13.LETOKWRI   1.16,
LETO16.LETOKWRI   1.16,
 LETO19.LETOKWRI   1.16/;

PARAMETER CAP(I,J)  aircraft capacity
 /EXXX10.KTIK10  25,
  KTIK11.CYQX11  25,
  CYQX12.EDAR13  25,
  EDAR13.EXXX13  25,
  KSUU11.KTIK11  50,
  KTIK12.KDOV12  50,
  KDOV14.EDAF15 146,
  EDAF18.KDOV19  50,
  KDOV21.KTIK21  75,
  KSUU14.KTIK14  50,
      .
      .
      .
      .
      .

  EDAR2 .EDAF2   18,
  EDAF10.EDAR10  18,
  EDAR10.EDAF10  18,
  EDAF17.EDAR17  18,
  EDAR17.EDAF17  18,
  KDOV1 .EDAR1   50,
  EDAR2 .LLBG3   50,
  LLBG3 .EDAR4   50,
  EDAR4 .KDOV5   50,
  KNGU20.LETO21  18/;

VARIABLE
Z  total delay

POSITIVE VARIABLES
X(I,J,K) shipment quantity
SUP(K)  total supply for each commodity K
DEL(K)  total amount delivered for each commodity
UNDEL(K)   amount not delivered for each commodity;
```

```
EQUATIONS
DELAY  objective function
SUMS(K)  total supply for each commodity K
SUPLY(IP,K)  conservation of flow for supply nodes
DEMND(IP,K)  conservation of flow for demand nodes
DELIVER(K)   amount delivered for each commodity
UNDELIVER(K) amount not delivered for each commodity
BAL(IP,K)  conservation of flow for ZIKN nodes
UB(I,J)  upper bound capac. constraint for aircraft;

DELAY .. Z =E= SUM((I,J,K)$E(I,J,K),
               C(I,J,K)*X(I,J,K));

SUMS(K) .. SUP(K) =E= SUM(I,S(I,K));

SUPLY(IP,K)$SIKN(IP,K)..SUM(J,X(IP,J,K)$E(IP,J,K)) -
                         SUM(I,X(I,IP,K)$E(I,IP,K))
                         =E= S(IP,K);

DEMND(IP,K)$DIKN(IP,K)..SUM(J,X(IP,J,K)$E(IP,J,K)) -
                         SUM(I,X(I,IP,K)$E(I,IP,K))
                         =L= SUP(K);

DELIVER(K) .. DEL(K) =E= SUM((I,IP)$E3(I,IP),
                         X(I,IP,K)$DIKN(IP,K));

UNDELIVER(K) .. UNDEL(K) =E= SUP(K) - DEL(K);

BAL(IP,K)$ZIKN(IP,K) .. SUM(J,X(IP,J,K)$E(IP,J,K)) -
                        SUM(I,X(I,IP,K)$E(I,IP,K))
                        =E= 0;

UB( E3(I,J) ) ..  SUM(K, X(I,J,K)) =L= CAP(E3);

MODEL MMCF /ALL/;

OPTION ITERLIM = 10000, RESLIM = 100000;
OPTION LIMROW = 0, LIMCOL = 0;

SOLVE MMCF USING LP MINIMIZING Z;
```

Appendix L: GAMS.TMP1 File

This appendix contains an extract of the "gams.tmp1" file which is created when the FORTRAN program, "GAMS.FOR" (shown in Appendix J), is executed. The file designates the airbase and time period for airbases serving as a supply node followed by the commodity (OD pair) which that airbase supplies.

```
EDAR1.EDARKNGU
EDAR4.EDARKNGU
EDAR7.EDARKNGU
EDAR10.EDARKNGU
EDAR13.EDARKNGU
EDAR16.EDARKNGU
EDAR19.EDARKNGU
EDAR1.EDARLGIR
EDAR4.EDARLGIR
EDAR7.EDARLGIR
    .
    .
    .
    .
    .

LETO13.LETOLIRN
LETO16.LETOLIRN
LETO19.LETOLIRN
LICZ1.LICZKSUU
LICZ4.LICZKSUU
LICZ7.LICZKSUU
LICZ10.LICZKSUU
LICZ13.LICZKSUU
LICZ16.LICZKSUU
LICZ19.LICZKSUU
```

Appendix M: GAMS.TMP2 File

This appendix contains an extract of the "gams.tmp2" file which is created when the FORTRAN program, "GAMS.FOR" (shown in Appendix J), is executed. The first column designates a mission leg (i.e., the starting airbase with time period and the ending airbase with time period), the second column shows the flight times in hours for that mission, and the third column designates the capacity of the aircraft.

```
EXXX10KTIK10  0.00   25
KTIK11CYQX11  4.74   25
CYQX12EDAR13  6.14   25
EDAR13EXXX13  0.00   25
KSUU11KTIK11  2.91   50
KTIK12KDOV12  2.81   50
KDOV14EDAF15  7.68  146
EDAF18KDOV19  9.60   50
KDOV21KTIK21  3.13   75
KSUU14KTIK14  2.91   50
     .
     .
     .
     .
     .
EDAR2 EDAF2   0.10   18
EDAF10EDAR10  0.10   18
EDAR10EDAF10  0.10   18
EDAF17EDAR17  0.10   18
EDAR17EDAF17  0.10   18
KDOV1 EDAR1   7.95   50
EDAR2 LLBG3   4.17   50
LLBG3 EDAR4   5.24   50
EDAR4 KDOV5   9.22   50
KNGU20LETO21  8.10   18
```

Appendix N: GAMS Program Output

This appendix contains an extract of the output from the GAMS program shown in Appendix K.

```
GAMS 2.20 VAX VMS
G E N E R A L   A L G E B R A I C   M O D E L I N G
S Y S T E M
C O M P I L A T I O N


SET K   COMMODITIES (CARGO)
 /EDARKNGU,
  EDARLGIR,
  EDARLICZ,
  EDARLIRN,
  EDAROEDR,
  EGUNKNGU,
  EGUNLTAG,
  KCHSEDAF,
  KDOVLGIR,
  KDOVLIPA,
  KDOVOEDR,
  KNGULIPA,
  KTIKLGIR,
  KTIKLIPA,
  KTIKLTAG,
  KTIKOEDR,
  KTIKOERY,
  LETOKDOV,
  LETOKTIK,
  LETOKWRI/;

SET I   AIRBASE-TIME PERIODS
 /BIKF1 * BIKF21,
  CYQX1 * CYQX21,
  EDAF1 * EDAF21,
  EDAR1 * EDAR21,
  EGUN1 * EGUN21,
  EXXX1 * EXXX21,
  FTTJ1 * FTTJ21,
  FZAA1 * FZAA21,
  GLRB1 * GLRB21,
  GOOY1 * GOOY21,
  HKNA1 * HKNA21,
  HSSS1 * HSSS21,
  KCHS1 * KCHS21,
  KDOV1 * KDOV21,
```

```
   KNGU1 * KNGU21,
   KSUU1 * KSUU21,
   KTIK1 * KTIK21,
   KWRI1 * KWRI21,
   KXXX1 * KXXX21,
   LCRA1 * LCRA21,
   LERT1 * LERT21,
   LETO1 * LETO21,
   LGIR1 * LGIR21,
   LICZ1 * LICZ21,
   LIPA1 * LIPA21,
   LIRN1 * LIRN21,
   LIRP1 * LIRP21,
   LLBG1 * LLBG21,
   LPLA1 * LPLA21,
   LTAG1 * LTAG21,
   OBBI1 * OBBI21,
   OEDR1 * OEDR21,
   OERY1 * OERY21,
   OJAF1 * OJAF21,
   OKBK1 * OKBK21,
   OMFJ1 * OMFJ21/;

ALIAS (I,IP);

ALIAS (I,J);

SET IK(I,K)  AIRBASE(AB)-CARGO COMBINATIONS
  /(BIKF1 * BIKF21).
  (EDARKNGU,
   EDARLGIR,
   EDARLICZ,
   EDARLIRN,
   EDAROEDR,
   EGUNKNGU,
   EGUNLTAG,
   KCHSEDAF,
   KDOVLGIR,
   KDOVLIPA,
   KDOVOEDR,
   KNGULIPA,
   KTIKLGIR,
   KTIKLIPA,
   KTIKLTAG,
   KTIKOEDR,
   KTIKOERY,
   LETOKDOV,
   LETOKTIK,
   LETOKWRI),
   (CYQX1 * CYQX21).
```

```
(EDARKNGU,
 EDARLGIR,
 EDARLICZ,
 EDARLIRN,
 EDAROEDR,
 EGUNKNGU,
 EGUNLTAG,
 KCHSEDAF,
 KDOVLGIR,
 KDOVLIPA,
 KDOVOEDR,
 KNGULIPA,
 KTIKLGIR,
 KTIKLIPA,
 KTIKLTAG,
 KTIKOEDR,
 KTIKOERY,
 LETOKDOV,
 LETOKTIK,
 LETOKWRI),
 (EDAF1 * EDAF21).
.
.
.
.
.

 (OKBK1 * OKBK21).
(EDARKNGU,
 EDARLGIR,
 EDARLICZ,
 EDARLIRN,
 EDAROEDR,
 EGUNKNGU,
 EGUNLTAG,
 KCHSEDAF,
 KDOVLGIR,
 KDOVLIPA,
 KDOVOEDR,
 KNGULIPA,
 KTIKLGIR,
 KTIKLIPA,
 KTIKLTAG,
 KTIKOEDR,
 KTIKOERY,
 LETOKDOV,
 LETOKTIK,
 LETOKWRI),
 (OMFJ1 * OMFJ21).
(EDARKNGU,
```

```
  EDARLGIR,
  EDARLICZ,
  EDARLIRN,
  EDAROEDR,
  EGUNKNGU,
  EGUNLTAG,
  KCHSEDAF,
  KDOVLGIR,
  KDOVLIPA,
  KDOVOEDR,
  KNGULIPA,
  KTIKLGIR,
  KTIKLIPA,
  KTIKLTAG,
  KTIKOEDR,
  KTIKOERY,
  LETOKDOV,
  LETOKTIK,
  LETOKWRI)/;

SET DIK(I,K)  DYNAMIC SET FOR IK;
DIK(I,K) = YES;

SET E1(I,J,K) ARCS FOR CARGO STAYING AT AB
 /(BIKF1.BIKF2, BIKF2.BIKF3, BIKF3.BIKF4,
 BIKF4.BIKF5, BIKF5.BIKF6, BIKF6.BIKF7,
 BIKF7.BIKF8, BIKF8.BIKF9, BIKF9.BIKF10,
 BIKF10.BIKF11, BIKF11.BIKF12, BIKF12.BIKF13,
 BIKF13.BIKF14, BIKF14.BIKF15, BIKF15.BIKF16,
 BIKF16.BIKF17, BIKF17.BIKF18, BIKF18.BIKF19,
 BIKF19.BIKF20, BIKF20.BIKF21, BIKF21.BIKF1,
 CYQX1.CYQX2, CYQX2.CYQX3, CYQX3.CYQX4,
 CYQX4.CYQX5, CYQX5.CYQX6, CYQX6.CYQX7,
 CYQX7.CYQX8, CYQX8.CYQX9, CYQX9.CYQX10,
 CYQX10.CYQX11, CYQX11.CYQX12, CYQX12.CYQX13,
 CYQX13.CYQX14, CYQX14.CYQX15, CYQX15.CYQX16,
 CYQX16.CYQX17, CYQX17.CYQX18, CYQX18.CYQX19,
 CYQX19.CYQX20, CYQX20.CYQX21, CYQX21.CYQX1,
 EDAF1.EDAF2, EDAF2.EDAF3, EDAF3.EDAF4,
 EDAF4.EDAF5, EDAF5.EDAF6, EDAF6.EDAF7,
 EDAF7.EDAF8, EDAF8.EDAF9, EDAF9.EDAF10,
 EDAF10.EDAF11, EDAF11.EDAF12, EDAF12.EDAF13,
 EDAF13.EDAF14, EDAF14.EDAF15, EDAF15.EDAF16,
 EDAF16.EDAF17, EDAF17.EDAF18, EDAF18.EDAF19,
 EDAF19.EDAF20, EDAF20.EDAF21, EDAF21.EDAF1,
.
.
.
.
.
```

```
 OKBK1.OKBK2, OKBK2.OKBK3, OKBK3.OKBK4,
 OKBK4.OKBK5, OKBK5.OKBK6, OKBK6.OKBK7,
 OKBK7.OKBK8, OKBK8.OKBK9, OKBK9.OKBK10,
 OKBK10.OKBK11, OKBK11.OKBK12, OKBK12.OKBK13,
 OKBK13.OKBK14, OKBK14.OKBK15, OKBK15.OKBK16,
 OKBK16.OKBK17, OKBK17.OKBK18, OKBK18.OKBK19,
 OKBK19.OKBK20, OKBK20.OKBK21, OKBK21.OKBK1,
 OMFJ1.OMFJ2, OMFJ2.OMFJ3, OMFJ3.OMFJ4,
 OMFJ4.OMFJ5, OMFJ5.OMFJ6, OMFJ6.OMFJ7,
 OMFJ7.OMFJ8, OMFJ8.OMFJ9, OMFJ9.OMFJ10,
 OMFJ10.OMFJ11, OMFJ11.OMFJ12, OMFJ12.OMFJ13,
 OMFJ13.OMFJ14, OMFJ14.OMFJ15, OMFJ15.OMFJ16,
 OMFJ16.OMFJ17, OMFJ17.OMFJ18, OMFJ18.OMFJ19,
 OMFJ19.OMFJ20, OMFJ20.OMFJ21, OMFJ21.OMFJ1).
 (EDARKNGU,
  EDARLGIR,
  EDARLICZ,
  EDARLIRN,
  EDAROEDR,
  EGUNKNGU,
  EGUNLTAG,
  KCHSEDAF,
  KDOVLGIR,
  KDOVLIPA,
  KDOVOEDR,
  KNGULIPA,
  KTIKLGIR,
  KTIKLIPA,
  KTIKLTAG,
  KTIKOEDR,
  KTIKOERY,
  LETOKDOV,
  LETOKTIK,
  LETOKWRI)/;

SET ET(I,J,K)  DYNAMIC SET FOR E1;
ET(I,J,K) = NO;
ET(E1) = YES;

SET E2(I,J,K) ARCS REPRESENTING A-C WITH CARGO
 /(EXXX10.KTIK10,
   KTIK11.CYQX11,
   CYQX12.EDAR13,
   EDAR13.EXXX13,
   KSUU11.KTIK11,
   KTIK12.KDOV12,
   KDOV14.EDAF15,
   EDAF18.KDOV19,
   KDOV21.KTIK21,
   KSUU14.KTIK14,
```

```
.
.
.
.
.
     EDAR2 .EDAF2 ,
     EDAF10.EDAR10,
     EDAR10.EDAF10,
     EDAF17.EDAR17,
     EDAR17.EDAF17,
     KDOV1 .EDAR1 ,
     EDAR2 .LLBG3 ,
     LLBG3 .EDAR4 ,
     EDAR4 .KDOV5 ,
     KNGU20.LETO21).
   (EDARKNGU,
    EDARLGIR,
    EDARLICZ,
    EDARLIRN,
    EDAROEDR,
    EGUNKNGU,
    EGUNLTAG,
    KCHSEDAF,
    KDOVLGIR,
    KDOVLIPA,
    KDOVOEDR,
    KNGULIPA,
    KTIKLGIR,
    KTIKLIPA,
    KTIKLTAG,
    KTIKOEDR,
    KTIKOERY,
    LETOKDOV,
    LETOKTIK,
    LETOKWRI)/;

   SET ES(I,J,K)  DYNAMIC SET FOR E2;
   ES(I,J,K) = NO;
   ES(E2) = YES;

   SET E(I,J,K) SET OF ALL ARCS (ET AND ES);
   E(I,J,K) = ET(I,J,K) + ES(I,J,K);

   SET E3(I,J)  ARCS REPRESENTING AIRCRAFT
   /EXXX10.KTIK10,
     KTIK11.CYQX11,
     CYQX12.EDAR13,
     EDAR13.EXXX13,
     KSUU11.KTIK11,
```

```
   KTIK12.KDOV12,
   KDOV14.EDAF15,
   EDAF18.KDOV19,
   KDOV21.KTIK21,
   KSUU14.KTIK14,
.
.
.
.
.

   EDAR2 .EDAF2 ,
   EDAF10.EDAR10,
   EDAR10.EDAF10,
   EDAF17.EDAR17,
   EDAR17.EDAF17,
   KDOV1 .EDAR1 ,
   EDAR2 .LLBG3 ,
   LLBG3 .EDAR4 ,
   EDAR4 .KDOV5 ,
   KNGU20.LETO21/;

SET SIKN(I,K) SUPPLY NODES FOR ALL CARGO
 /EDAR1.EDARKNGU, EDAR4.EDARKNGU, EDAR7.EDARKNGU,
 EDAR10.EDARKNGU, EDAR13.EDARKNGU, EDAR16.EDARKNGU,
 EDAR19.EDARKNGU,
 EDAR1.EDARLGIR, EDAR4.EDARLGIR, EDAR7.EDARLGIR,
 EDAR10.EDARLGIR, EDAR13.EDARLGIR, EDAR16.EDARLGIR,
 EDAR19.EDARLGIR,
 EDAR1.EDARLICZ, EDAR4.EDARLICZ, EDAR7.EDARLICZ,
 EDAR10.EDARLICZ, EDAR13.EDARLICZ, EDAR16.EDARLICZ,
 EDAR19.EDARLICZ,
 EDAR1.EDARLIRN, EDAR4.EDARLIRN, EDAR7.EDARLIRN,
 EDAR10.EDARLIRN, EDAR13.EDARLIRN, EDAR16.EDARLIRN,
 EDAR19.EDARLIRN,
.
.
.
.
.

 KTIK1.KTIKOERY, KTIK4.KTIKOERY, KTIK7.KTIKOERY,
 KTIK10.KTIKOERY, KTIK13.KTIKOERY, KTIK16.KTIKOERY,
 KTIK19.KTIKOERY,
 LETO1.LETOKDOV, LETO4.LETOKDOV, LETO7.LETOKDOV,
 LETO10.LETOKDOV, LETO13.LETOKDOV, LETO16.LETOKDOV,
 LETO19.LETOKDOV,
 LETO1.LETOKTIK, LETO4.LETOKTIK, LETO7.LETOKTIK,
 LETO10.LETOKTIK, LETO13.LETOKTIK, LETO16.LETOKTIK,
 LETO19.LETOKTIK,
```

```
 LETO1.LETOKWRI, LETO4.LETOKWRI, LETO7.LETOKWRI,
 LETO10.LETOKWRI, LETO13.LETOKWRI, LETO16.LETOKWRI,
 LETO19.LETOKWRI/;

SET SUPNODE(I,K)  DYNAMIC SET FOR SIKN;
SUPNODE(I,K) = NO;
SUPNODE(SIKN) = YES;

SET DIKN(I,K) DEMAND NODES FOR ALL CARGO
 /(KNGU1 * KNGU21).EDARKNGU,
 (LGIR1 * LGIR21).EDARLGIR,
 (LICZ1 * LICZ21).EDARLICZ,
 (LIRN1 * LIRN21).EDARLIRN,
 (OEDR1 * OEDR21).EDAROEDR,
 (KNGU1 * KNGU21).EGUNKNGU,
 (LTAG1 * LTAG21).EGUNLTAG,
 (EDAF1 * EDAF21).KCHSEDAF,
 (LGIR1 * LGIR21).KDOVLGIR,
 (LIPA1 * LIPA21).KDOVLIPA,
 (OEDR1 * OEDR21).KDOVOEDR,
 (LIPA1 * LIPA21).KNGULIPA,
 (LGIR1 * LGIR21).KTIKLGIR,
 (LIPA1 * LIPA21).KTIKLIPA,
 (LTAG1 * LTAG21).KTIKLTAG,
 (OEDR1 * OEDR21).KTIKOEDR,
 (OERY1 * OERY21).KTIKOERY,
 (KDOV1 * KDOV21).LETOKDOV,
 (KTIK1 * KTIK21).LETOKTIK,
 (KWRI1 * KWRI21).LETOKWRI/;

SET DMDNODE(I,K)  DYNAMIC SET FOR DIKN;
DMDNODE(I,K) = NO;
DMDNODE(DIKN) = YES;

SET ZIKN(I,K)  NEITHER DEMAND NOR SUPPLY NODES;
ZIKN(I,K) = DIK(I,K) - SUPNODE(I,K) - DMDNODE(I,K);

PARAMETER C(I,J,K)  DELAY;

C(I,J,K) = 0;

C(I,J,K)$ET(I,J,K) = 8;

 C("EXXX10","KTIK10",K)=  0.0;
 C("KTIK11","CYQX11",K)=  4.7;
 C("CYQX12","EDAR13",K)=  6.1;
 C("EDAR13","EXXX13",K)=  0.0;
 C("KSUU11","KTIK11",K)=  2.9;
 C("KTIK12","KDOV12",K)=  2.8;
 C("KDOV14","EDAF15",K)=  7.7;
```

```
C("EDAF18","KDOV19",K)=  9.6;
C("KDOV21","KTIK21",K)=  3.1;
C("KSUU14","KTIK14",K)=  2.9;
 .
 .
 .
 .
 .
C("EDAR2 ","EDAF2 ",K)=  0.1;
C("EDAF10","EDAR10",K)=  0.1;
C("EDAR10","EDAF10",K)=  0.1;
C("EDAF17","EDAR17",K)=  0.1;
C("EDAR17","EDAF17",K)=  0.1;
C("KDOV1 ","EDAR1 ",K)=  7.9;
C("EDAR2 ","LLBG3 ",K)=  4.2;
C("LLBG3 ","EDAR4 ",K)=  5.2;
C("EDAR4 ","KDOV5 ",K)=  9.2;
C("KNGU20","LETO21",K)=  8.1;

PARAMETER S(I,K) THE SUPPLY AT NODE SIKN
 /EDAR1.EDARKNGU   0.24, EDAR4.EDARKNGU   0.24,
 EDAR7.EDARKNGU   0.24,
 EDAR10.EDARKNGU   0.24, EDAR13.EDARKNGU   0.24,
 EDAR16.EDARKNGU   0.24,
 EDAR19.EDARKNGU   0.24,
 EDAR1.EDARLGIR   0.30, EDAR4.EDARLGIR   0.29,
 EDAR7.EDARLGIR   0.30,
 EDAR10.EDARLGIR   0.30, EDAR13.EDARLGIR   0.29,
 EDAR16.EDARLGIR   0.30,
 EDAR19.EDARLGIR   0.30,
 EDAR1.EDARLICZ   0.18, EDAR4.EDARLICZ   0.18,
 EDAR7.EDARLICZ   0.18,
 EDAR10.EDARLICZ   0.18, EDAR13.EDARLICZ   0.18,
 EDAR16.EDARLICZ   0.18,
 EDAR19.EDARLICZ   0.18,
 EDAR1.EDARLIRN   0.18, EDAR4.EDARLIRN   0.19,
 EDAR7.EDARLIRN   0.18,
 EDAR10.EDARLIRN   0.18, EDAR13.EDARLIRN   0.19,
 EDAR16.EDARLIRN   0.18,
 EDAR19.EDARLIRN   0.18,
 .
 .
 .
 .
 .
 KTIK1.KTIKOERY   0.50, KTIK4.KTIKOERY   0.25,
 KTIK7.KTIKOERY   0.14,
 KTIK10.KTIKOERY   0.53, KTIK13.KTIKOERY   0.84,
```

```
  KTIK16.KTIKOERY   0.81,
  KTIK19.KTIKOERY   0.80,
  LETO1.LETOKDOV   8.19, LETO4.LETOKDOV   8.18,
  LETO7.LETOKDOV   8.19,
  LETO10.LETOKDOV   8.19, LETO13.LETOKDOV   8.18,
  LETO16.LETOKDOV   8.19,
  LETO19.LETOKDOV   8.19,
  LETO1.LETOKTIK   0.77, LETO4.LETOKTIK   0.77,
  LETO7.LETOKTIK   0.77,
  LETO10.LETOKTIK   0.77, LETO13.LETOKTIK   0.77,
  LETO16.LETOKTIK   0.77,
  LETO19.LETOKTIK   0.77,
  LETO1.LETOKWRI   1.16, LETO4.LETOKWRI   1.16,
  LETO7.LETOKWRI   1.16,
  LETO10.LETOKWRI   1.16, LETO13.LETOKWRI   1.16,
  LETO16.LETOKWRI   1.16,
  LETO19.LETOKWRI   1.16/;

PARAMETER CAP(I,J)  AIRCRAFT CAPACITY
 /EXXX10.KTIK10  25,
  KTIK11.CYQX11  25,
  CYQX12.EDAR13  25,
  EDAR13.EXXX13  25,
  KSUU11.KTIK11  50,
  KTIK12.KDOV12  50,
  KDOV14.EDAF15 146,
  EDAF18.KDOV19  50,
  KDOV21.KTIK21  75,
  KSUU14.KTIK14  50,
.
.
.
.
.
  EDAR2 .EDAF2   18,
  EDAF10.EDAR10  18,
  EDAR10.EDAF10  18,
  EDAF17.EDAR17  18,
  EDAR17.EDAF17  18,
  KDOV1 .EDAR1   50,
  EDAR2 .LLBG3   50,
  LLBG3 .EDAR4   50,
  EDAR4 .KDOV5   50,
  KNGU20.LETO21  18/;

VARIABLE
Z  TOTAL DELAY

POSITIVE VARIABLES
```

```
X(I,J,K) SHIPMENT QUANTITY
SUP(K)  TOTAL SUPPLY FOR EACH CARGO K
DEL(K)  TOTAL AMOUNT DELIVERED FOR EACH CARGO
UNDEL(K)   AMOUNT NOT DELIVERED FOR EACH CARGO;

EQUATIONS
DELAY  OBJECTIVE FUNCTION
SUMS(K)  TOTAL SUPPLY FOR EACH CARGO K
SUPLY(IP,K)  CONSERVATION OF FLOW FOR SUPPLY NODES
DEMND(IP,K)  CONSERVATION OF FLOW FOR DEMAND NODES
DELIVER(K)   AMOUNT DELIVERED FOR EACH CARGO
UNDELIVER(K) AMOUNT NOT DELIVERED FOR EACH CARGO
BAL(IP,K)  CONSERVATION OF FLOW FOR ZIKN NODES
UB(I,J)  UPPER BOUND CAPAC. CONSTRAINT FOR AIRCRAFT;

DELAY .. Z =E= SUM((I,J,K)$E(I,J,K),
               C(I,J,K)*X(I,J,K));

SUMS(K) .. SUP(K) =E= SUM(I,S(I,K));

SUPLY(IP,K)$SIKN(IP,K)..SUM(J,X(IP,J,K)$E(IP,J,K)) -
                         SUM(I,X(I,IP,K)$E(I,IP,K))
                         =E= S(IP,K);

DEMND(IP,K)$DIKN(IP,K)..SUM(J,X(IP,J,K)$E(IP,J,K)) -
                         SUM(I,X(I,IP,K)$E(I,IP,K))
                         =L= SUP(K);

DELIVER(K) .. DEL(K) =E= SUM((I,IP)$E3(I,IP),
                         X(I,IP,K)$DIKN(IP,K));

UNDELIVER(K) .. UNDEL(K) =E= SUP(K) - DEL(K);

BAL(IP,K)$ZIKN(IP,K) .. SUM(J,X(IP,J,K)$E(IP,J,K)) -
                        SUM(I,X(I,IP,K)$E(I,IP,K))
                        =E= 0;

UB( E3(I,J) ) ..  SUM(K, X(I,J,K)) =L= CAP(E3);

MODEL MMCF /ALL/;

OPTION ITERLIM = 10000, RESLIM = 100000;
OPTION LIMROW = 0, LIMCOL = 0;

SOLVE MMCF USING LP MINIMIZING Z;
  .
  .
  .
  .
  .
```

SETS

DIK	DYNAMIC SET FOR IK
DIKN	DEMAND NODES FOR ALL CARGO
DMDNODE	DYNAMIC SET FOR DIKN
E	SET OF ALL ARCS (ET AND ES)
E1	ARCS FOR CARGO STAYING AT AB
E2	ARCS REPRESENTING A-C WITH CARGO
E3	ARCS REPRESENTING AIRCRAFT
ES	DYNAMIC SET FOR E2
ET	DYNAMIC SET FOR E1
I	AIRBASE-TIME PERIODS
IK	AIRBASE(AB)-CARGO COMBINATIONS
IP	ALIASED WITH I
J	ALIASED WITH I
K	COMMODITIES (CARGO)
SIKN	SUPPLY NODES FOR ALL CARGO
SUPNODE	DYNAMIC SET FOR SIKN
ZIKN	NEITHER DEMAND NOR SUPPLY NODES

PARAMETERS

C	DELAY
CAP	AIRCRAFT CAPACITY
S	THE SUPPLY AT NODE SIKN

VARIABLES

DEL	TOTAL AMOUNT DELIVERED FOR EACH CARGO
SUP	TOTAL SUPPLY FOR EACH CARGO K
UNDEL	AMOUNT NOT DELIVERED FOR EACH CARGO
X	SHIPMENT QUANTITY
Z	TOTAL DELAY

EQUATIONS

BAL	CONSERVATION OF FLOW FOR ZIKN NODES
DELAY	OBJECTIVE FUNCTION
DELIVER	AMOUNT DELIVERED FOR EACH CARGO
DEMND	CONSERVATION OF FLOW FOR DEMAND NODES
SUMS	TOTAL SUPPLY FOR EACH CARGO K
SUPLY	CONSERVATION OF FLOW FOR SUPPLY NODES
UB	UPPER BOUND CAPAC. CONSTRAINT FOR AIRCRAFT
UNDELIVER	AMOUNT NOT DELIVERED FOR EACH CARGO

MODELS

```
MMCF

COMPILATION TIME       =        3.040 SECONDS

MODEL STATISTICS

BLOCKS OF EQUATIONS      8      SINGLE EQUATIONS  15422
BLOCKS OF VARIABLES      5      SINGLE VARIABLES  20001
NON ZERO ELEMENTS    65246

GENERATION TIME        =      188.560 SECONDS

EXECUTION TIME         =      199.490 SECONDS

SOLUTION REPORT      SOLVE MMCF USING LP FROM LINE 2313
```

S O L V E   S U M M A R Y

MODEL	MMCF	OBJECTIVE	Z
TYPE	LP	DIRECTION	MINIMIZE
SOLVER	MINOS5	FROM LINE	2313

**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS 1 OPTIMAL
**** OBJECTIVE VALUE 9301.0290

RESOURCE USAGE, LIMIT	12475.550	100000.000
ITERATION COUNT, LIMIT	8688	10000

M I N O S 5.2 (Mar 1988)
= = = = =

B. A. Murtagh, University of New South Wales
and
P. E. Gill, W. Murray, M. A. Saunders and M. H. Wright
Systems Optimization Laboratory, Stanford University.

Work space needed (estimate) -- 786945 words.
Work space available -- 944335 words.

EXIT -- OPTIMAL SOLUTION FOUND

	LOWER	LEVEL	UPPER	MARGINAL
---- EQU DELAY	.	.	.	1.000

DELAY OBJECTIVE FUNCTION

---- EQU SUMS TOTAL SUPPLY FOR EACH CARGO K

	LOWER	LEVEL	UPPER	MARGINAL
EDARKNGU	1.680	1.680	1.680	EPS
EDARLGIR	2.080	2.080	2.080	EPS
EDARLICZ	1.260	1.260	1.260	EPS
EDARLIRN	1.280	1.280	1.280	EPS
EDAROEDR	5.930	5.930	5.930	EPS
EGUNKNGU	5.460	5.460	5.460	EPS
EGUNLTAG	11.760	11.760	11.760	EPS
KCHSEDAF	1.240	1.240	1.240	EPS

KDOVLGIR	2.120	2.120	2.120	EPS
KDOVLIPA	42.580	42.580	42.580	EPS
KDOVOEDR	42.750	42.750	42.750	EPS
KNGULIPA	10.500	10.500	10.500	EPS
KTIKLGIR	1.870	1.870	1.870	EPS
KTIKLIPA	3.940	3.940	3.940	EPS
KTIKLTAG	6.390	6.390	6.390	EPS
KTIKOEDR	7.230	7.230	7.230	EPS
KTIKOERY	3.870	3.870	3.870	EPS
LETOKDOV	57.310	57.310	57.310	EPS
LETOKTIK	5.390	5.390	5.390	EPS
LETOKWRI	8.120	8.120	8.120	EPS

---- EQU SUPLY CONSERVATION OF FLOW FOR SUPPLY NODES

	LOWER	LEVEL	UPPER	MARGINAL
EDAR1 .EDARKNGU	0.240	0.240	0.240	72.100
EDAR1 .EDARLGIR	0.300	0.300	0.300	70.300
EDAR1 .EDARLICZ	0.180	0.180	0.180	52.600
EDAR1 .EDARLIRN	0.180	0.180	0.180	43.600
EDAR1 .EDAROEDR	0.850	0.850	0.850	22.700
EDAR4 .EDARKNGU	0.240	0.240	0.240	54.700
EDAR4 .EDARLGIR	0.290	0.290	0.290	63.900
EDAR4 .EDARLICZ	0.180	0.180	0.180	35.200
EDAR4 .EDARLIRN	0.190	0.190	0.190	26.200
EDAR4 .EDAROEDR	0.840	0.840	0.840	42.300
.				
.				
.				
.				
.				
LETO10.LETOKWRI	1.160	1.160	1.160	43.900
LETO13.LETOKDOV	8.180	8.180	8.180	41.700
LETO13.LETOKTIK	.770	0.770	0.770	63.800

LETO13.LETOKWRI	1.160	1.160	1.160	130.700
LETO16.LETOKDOV	8.190	8.190	8.190	17.700
LETO16.LETOKTIK	0.770	0.770	0.770	39.800
LETO16.LETOKWRI	1.160	1.160	1.160	106.700
LETO19.LETOKDOV	8.190	8.190	8.190	56.400
LETO19.LETOKTIK	0.770	0.770	0.770	121.000
LETO19.LETOKWRI	1.160	1.160	1.160	130.100

---- EQU DEMND CONSERVATION OF FLOW FOR DEMAND NODES

	LOWER	LEVEL	UPPER	MARGINAL
EDAF1 .KCHSEDAF	-INF	-1.240	.	.
EDAF2 .KCHSEDAF	-INF	-1.240	.	.
EDAF3 .KCHSEDAF	-INF	-1.240	.	.
EDAF4 .KCHSEDAF	-INF	-1.890	.	.
EDAF5 .KCHSEDAF	-INF	-1.240	.	.
EDAF6 .KCHSEDAF	-INF	-1.240	.	.
EDAF7 .KCHSEDAF	-INF	-1.240	.	.
EDAF8 .KCHSEDAF	-INF	-1.240	.	.
EDAF9 .KCHSEDAF	-INF	-1.280	.	.
EDAF10.KCHSEDAF	-INF	-1.240	.	.

.
.
.
.
.

OERY12.KTIKOERY	-INF	-3.870	.	.
OERY13 KTIKOERY	-INF	-3.870	.	.

OERY14.KTIKOERY	-INF	-3.870	.	.
OERY15.KTIKOERY	-INF	-3.870	.	.
OERY16.KTIKOERY	-INF	-5.290	.	.
OERY17.KTIKOERY	-INF	-3.870	.	.
OERY18.KTIKOERY	-INF	-3.870	.	.
OERY19.KTIKOERY	-INF	-3.870	.	.
OERY20.KTIKOERY	-INF	-5.520	.	.
OERY21.KTIKOERY	-INF	-3.870	.	.

---- EQU UNDELIVER AMOUNT NOT DELIVERED FOR EACH CARGO

	LOWER	LEVEL	UPPER	MARGINAL
EDARKNGU	.	.	.	EPS
EDARLGIR	.	.	.	EPS
EDARLICZ	.	.	.	EPS
EDARLIRN	.	.	.	EPS
EDAROEDR	.	.	.	EPS
EGUNKNGU	.	.	.	EPS
EGUNLTAG	.	.	.	EPS
KCHSEDAF	.	.	.	EPS
KDOVLGIR	.	.	.	EPS
KDOVLIPA	.	.	.	EPS
KDOVOEDR	.	.	.	EPS
KNGULIPA	.	.	.	EPS
KTIKLGIR	.	.	.	EPS
KTIKLIPA	.	.	.	EPS
KTIKLTAG	.	.	.	EPS
KTIKOEDR	.	.	.	EPS
KTIKOERY	.	.	.	EPS
LETOKDOV	.	.	.	EPS
LETOKTIK	.	.	.	EPS
LETOKWRI	.	.	.	EPS

---- EQU BAL CONSERVATION OF FLOW FOR ZIKN NODES

	LOWER	LEVEL	UPPER	MARGINAL
BIKF1 .EDARKNGU	.	.	.	2.500
BIKF1 .EDARLGIR	.	.	.	12.000
BIKF1 .EDARLICZ	.	.	.	5.800
BIKF1 .EDARLIRN	.	.	.	9.000
BIKF1 .EDAROEDR	.	.	.	-17.800
BIKF1 .EGUNKNGU	.	.	.	2.500
BIKF1 .EGUNLTAG	.	.	.	-17.600
BIKF1 .KCHSEDAF	.	.	.	33.800
BIKF1 .KDOVLGIR	.	.	.	80.000
BIKF1 .KDOVLIPA	.	.	.	55.700
.				
.				
.				
.				
.				
OMFJ21.KDOVOEDR	.	.	.	-29.700
OMFJ21.KNGULIPA	.	.	.	-33.800
OMFJ21.KTIKLGIR	.	.	.	15.100
OMFJ21.KTIKLIPA	.	.	.	-37.100
OMFJ21.KTIKLTAG	.	.	.	-26.100
OMFJ21.KTIKOEDR	.	.	.	-29.900
OMFJ21.KTIKOERY	.	.	.	-15.600
OMFJ21.LETOKDOV	.	.	.	-33.800
OMFJ21.LETOKTIK	.	.	.	-3.600
OMFJ21.LETOKWRI	.	.	.	-46.600

```
---- EQU UB          UPPER BOUND CAPAC. CONSTRAINT FOR AIRCRAFT

                     LOWER     LEVEL     UPPER    MARGINAL

BIKF4 .EGUN4         -INF      3.640    25.000       .
CYQX12.EDAR13        -INF        .      25.000       .
EDAF1 .LIRN1         -INF      0.720    18.000       .
EDAF2 .EDAR2         -INF      8.210    18.000       .
EDAF3 .OEDR4         -INF     19.180    71.000       .
EDAF4 .LETO4         -INF        .      18.000       .
EDAF4 .LIPA5         -INF      7.000     7.000    -24.400
EDAF4 .LTAG4         -INF      5.040    18.000       .
EDAF4 .OEDR5         -INF        .      18.000       .
EDAF4 .OKBK5         -INF      0.800    18.000       .
      .
      .
      .
      .
      .

OEDR17.EDAF18        -INF        .      18.000       .
OEDR20.OERY20        -INF      1.650    18.000       .
OERY8 .EDAF9         -INF        .      18.000       .
OERY18.EDAF19        -INF        .      25.000       .
OJAF13.EDAR14        -INF        .      18.000       .
OKBK5 .OEDR5         -INF      0.800    18.000       .
OKBK18.OEDR18        -INF      8.260    18.000       .
OMFJ10.OBBI10        -INF        .      18.000       .
OMFJ12.OBBI12        -INF        .      18.000       .
OMFJ19.OBBI19        -INF        .      18.000       .

                           LOWER     LEVEL     UPPER    MARGINAL

---- VAR Z                 -INF    9301.029    +INF        .

  Z           TOTAL DELAY

---- VAR X          SHIPMENT QUANTITY

                              LOWER     LEVEL     UPPER    MARGINAL

BIKF1 .BIKF2 .EDARKNGU          .         .       +INF        .
BIKF1 .BIKF2 .EDARLGIR          .         .       +INF        .
BIKF1 .BIKF2 .EDARLICZ          .         .       +INF        .
BIKF1 .BIKF2 .EDARLIRN          .         .       +INF        .
BIKF1 .BIKF2 .EDAROEDR          .         .       +INF        .
BIKF1 .BIKF2 .EGUNKNGU          .         .       +INF        .
BIKF1 .BIKF2 .EGUNLTAG          .         .       +INF        .
```

```
BIKF1 .BIKF2 .KCHSEDAF      .          .       +INF       .
BIKF1 .BIKF2 .KDOVLGIR      .          .       +INF       .
BIKF1 .BIKF2 .KDOVLIPA      .          .       +INF       .
      .
      .
      .
      .
      .
BIKF2 .BIKF3 .EDARKNGU      .          .       +INF    42.500
BIKF2 .BIKF3 .EDARLGIR      .          .       +INF    68.000
BIKF2 .BIKF3 .EDARLICZ      .          .       +INF    57.800
BIKF2 .BIKF3 .EDARLIRN      .          .       +INF    44.300
BIKF2 .BIKF3 .EDAROEDR      .          .       +INF    57.100
BIKF2 .BIKF3 .EGUNKNGU      .          .       +INF    42.500
BIKF2 .BIKF3 .EGUNLTAG      .          .       +INF    55.800
BIKF2 .BIKF3 .KCHSEDAF      .        0.650     +INF       .
BIKF2 .BIKF3 .KDOVLGIR      .          .       +INF       .
BIKF2 .BIKF3 .KDOVLIPA      .          .       +INF     5.100
BIKF2 .BIKF3 .KDOVOEDR      .          .       +INF     9.200
BIKF2 .BIKF3 .KNGULIPA      .        2.990     +INF       .
      .
      .
      .
BIKF3 .BIKF4 .KCHSEDAF      .        0.650     +INF       .
BIKF3 .BIKF4 .KDOVLGIR      .          .       +INF       .
BIKF3 .BIKF4 .KDOVLIPA      .          .       +INF       .
BIKF3 .BIKF4 .KDOVOEDR      .          .       +INF       .
BIKF3 .BIKF4 .KNGULIPA      .        2.990     +INF       .
      .
      .
      .
      .
      .

      .
      .
      .
      .
      .
EDAF1 .EDAF2 .EDAROEDR      .        0.850     +INF       .
EDAF1 .EDAF2 .EGUNKNGU      .          .       +INF       .
EDAF1 .EDAF2 .EGUNLTAG      .        1.680     +INF       .
EDAF1 .EDAF2 .KCHSEDAF      .          .       +INF     8.000
EDAF1 .EDAF2 .KDOVLGIR      .          .       +INF       .
EDAF1 .EDAF2 .KDOVLIPA      .        9.860     +INF       .
EDAF1 .EDAF2 .KDOVOEDR      .       15.970     +INF       .
EDAF1 .EDAF2 .KNGULIPA      .          .       +INF       .
EDAF1 .EDAF2 .KTIKLGIR      .        0.390     +INF       .
EDAF1 .EDAF2 .KTIKLIPA      .        1.670     +INF       .
```

EDAF1 .EDAF2 .KTIKLTAG	.	.	+INF	.
EDAF1 .EDAF2 .KTIKOEDR	.	1.510	+INF	.
EDAF1 .EDAF2 .KTIKOERY	.	0.800	+INF	.
EDAF1 .EDAF2 .LETOKDOV	.	.	+INF	.
EDAF1 .EDAF2 .LETOKTIK	.	.	+INF	.
EDAF1 .EDAF2 .LETOKWRI	.	.	+INF	.
EDAF1 .LIRN1 .EDARKNGU	.	.	+INF	.
EDAF1 .LIRN1 .EDARLGIR	.	.	+INF	.
EDAF1 .LIRN1 .EDARLICZ	.	0.360	+INF	.
EDAF1 .LIRN1 .EDARLIRN	.	0.360	+INF	.

.
.
.
.
.

.
.
.
.
.

OMFJ21.OMFJ1 .KDOVOEDR	.	.	+INF	45.700
OMFJ21.OMFJ1 .KNGULIPA	.	.	+INF	49.800
OMFJ21.OMFJ1 .KTIKLGIR	.	.	+INF	0.900
OMFJ21.OMFJ1 .KTIKLIPA	.	.	+INF	53.100
OMFJ21.OMFJ1 .KTIKLTAG	.	.	+INF	.
OMFJ21.OMFJ1 .KTIKOEDR	.	.	+INF	94.700
OMFJ21.OMFJ1 .KTIKOERY	.	.	+INF	31.600
OMFJ21.OMFJ1 .LETOKDOV	.	.	+INF	49.800
OMFJ21.OMFJ1 .LETOKTIK	.	.	+INF	.
OMFJ21.OMFJ1 .LETOKWRI	.	.	+INF	62.600

---- VAR SUP TOTAL SUPPLY FOR EACH CARGO K

	LOWER	LEVEL	UPPER	MARGINAL
EDARKNGU	.	1.680	+INF	.
EDARLGIR	.	2.080	+INF	.
EDARLICZ	.	1.260	+INF	.
EDARLIRN	.	1.280	+INF	.
EDAROEDR	.	5.930	+INF	.
EGUNKNGU	.	5.460	+INF	.
EGUNLTAG	.	11.760	+INF	.
KCHSEDAF	.	1.240	+INF	.
KDOVLGIR	.	2.120	+INF	.
KDOVLIPA	.	42.580	+INF	.
KDOVOEDR	.	42.750	+INF	.

KNGULIPA	.	10.500	+INF	.
KTIKLGIR	.	1.870	+INF	.
KTIKLIPA	.	3.940	+INF	.
KTIKLTAG	.	6.390	+INF	.
KTIKOEDR	.	7.230	+INF	.
KTIKOERY	.	3.870	+INF	.
LETOKDOV	.	57.310	+INF	.
LETOKTIK	.	5.390	+INF	.
LETOKWRI	.	8.120	+INF	.

---- VAR DEL TOTAL AMOUNT DELIVERED FOR EACH CARGO

	LOWER	LEVEL	UPPER	MARGINAL
EDARKNGU	.	1.680	+INF	.
EDARLGIR	.	2.080	+INF	.
EDARLICZ	.	1.260	+INF	.
EDARLIRN	.	1.280	+INF	.
EDAROEDR	.	5.930	+INF	.
EGUNKNGU	.	5.460	+INF	.
EGUNLTAG	.	11.760	+INF	.
KCHSEDAF	.	1.240	+INF	.
KDOVLGIR	.	2.120	+INF	.
KDOVLIPA	.	42.580	+INF	.
KDOVOEDR	.	42.750	+INF	.
KNGULIPA	.	10.500	+INF	.
KTIKLGIR	.	1.870	+INF	.
KTIKLIPA	.	3.940	+INF	.
KTIKLTAG	.	6.390	+INF	.
KTIKOEDR	.	7.230	+INF	.
KTIKOERY	.	3.870	+INF	.
LETOKDOV	.	57.310	+INF	.
LETOKTIK	.	5.390	+INF	.
LETOKWRI	.	8.120	+INF	.

---- VAR UNDEL AMOUNT NOT DELIVERED FOR EACH CARGO

	LOWER	LEVEL	UPPER	MARGINAL
EDARKNGU	.	.	+INF	.
EDARLGIR	.	.	+INF	.
EDARLICZ	.	.	+INF	.
EDARLIRN	.	.	+INF	.
EDAROEDR	.	.	+INF	.
EGUNKNGU	.	.	+INF	.
EGUNLTAG	.	.	+INF	.
KCHSEDAF	.	.	+INF	.
KDOVLGIR	.	.	+INF	.

```
KDOVLIPA        .         .        +INF        .
KDOVOEDR        .         .        +INF        .
KNGULIPA        .         .        +INF        .
KTIKLGIR        .         .        +INF        .
KTIKLIPA        .         .        +INF        .
KTIKLTAG        .         .        +INF        .
KTIKOEDR        .         .        +INF        .
KTIKOERY        .         .        +INF        .
LETOKDOV        .         .        +INF        .
LETOKTIK        .         .        +INF        .
LETOKWRI        .         .        +INF        .

**** REPORT SUMMARY :          0     NONOPT
                               0 INFEASIBLE
                               0  UNBOUNDED

**** FILE SUMMARY

INPUT    GOR93M:[MDELROSA]EXAMP2.GMS;13
OUTPUT   GOR93M:[MDELROSA]EXAMP2.LIS;13

EXECUTION TIME       =        34.180 SECONDS
```

Appendix O: Post-processed Data

This appendix contains the post-processed results from the subproblem in Chapter III. This appendix contains only the nonzero variables representing mission legs. The columns in the table below show the mission leg (designated by the four letter ICAO code and the time period), the average flight time of the aircraft in hours, the capacity of the aircraft (CAP), the cargo being carried on the aircraft (OD Pair), the amount of that cargo (QTY DEL.), and the mission numbers associated with the mission leg.

MISSION LEG	AVG FLT TIME	CAP	OD PAIR	QTY DEL.	MISSION NUMBER
EDAF10.KCHS11	10.60	18	EDARKNGU	0.960	59
EDAR16.KDOV17	9.50	18	EDARKNGU	0.240	200
EGUN4 .KCHS6	9.02	25	EDARKNGU	0.240	202
KCHS11.KNGU11	1.10	18	EDARKNGU	0.960	216
EDAR20.EGUN21	1.50	18	EDARKNGU	0.240	230
EDAR8 .EGUN8	1.50	18	EDARKNGU	0.720	230
EGUN8 .EDAF9	1.50	18	EDARKNGU	0.720	230
EDAR14.KDOV15	8.74	30	EDARKNGU	0.240	252
KDOV17.KNGU17	0.80	18	EDARKNGU	0.480	255
KCHS7 .KNGU7	1.05	75	EDARKNGU	0.240	259
EDAR10.EDAF10	0.10	18	EDARKNGU	0.240	292
EDAR2 .LLBG3	4.17	50	EDARKNGU	0.240	293
LLBG3 .EDAR4	5.24	50	EDARKNGU	0.240	293
EDAF10.LETO11	2.60	18	EDARLGIR	0.600	230
EDAR8 .EGUN8	1.50	18	EDARLGIR	0.300	230
EGUN8 .EDAF9	1.50	18	EDARLGIR	0.300	230
EDAR17.LIPA17	1.50	18	EDARLGIR	0.590	231
EDAF12.LTAG13	4.40	18	EDARLGIR	0.600	237
LTAG13.EDAF14	5.20	18	EDARLGIR	0.600	237
EDAF14.LIPA15	2.64	7	EDARLGIR	0.600	251
LIPA17.LGIR17	3.20	7	EDARLGIR	1.190	251
LTAG20.LCRA20	1.81	7	EDARLGIR	0.300	251
LCRA21.LGIR21	1.95	7	EDARLGIR	0.300	251
EDAF4 .LIPA5	2.64	7	EDARLGIR	0.300	251
LIPA7 .LGIR7	3.20	7	EDARLGIR	0.300	251
LTAG10.LCRA10	1.81	7	EDARLGIR	0.290	251
LCRA11.LGIR11	1.95	7	EDARLGIR	0.290	251
EDAR19.LTAG19	4.14	30	EDARLGIR	0.300	252
EDAR5 .LTAG5	4.14	30	EDARLGIR	0.290	252
LETO11.EDAF11	2.60	18	EDARLGIR	0.600	262
EDAR2 .EDAF2	0.10	18	EDARLGIR	0.300	292
EDAR10.EDAF10	0.10	18	EDARLGIR	0.300	292

EDAR7 .EDAF7	0.10	18	EDARLIRN	0.550	59
EDAR16.EDAF16	0.10	18	EDARLIRN	0.180	59
EDAF10.LETO11	2.60	18	EDARLIRN	0.180	230
EDAR14.EGUN14	1.50	18	EDARLIRN	0.190	230
EDAF17.LETO17	2.60	18	EDARLIRN	.180	230
LETO17.LIPA18	2.20	18	EDARLIRN	0.180	230
LIPA20.EDAR20	1.80	18	EDARLIRN	0.180	230
EDAR20.EGUN21	1.50	18	EDARLIRN	0.360	230
EGUN21.EDAF21	1.50	18	EDARLIRN	0.360	230
EDAF12.LTAG13	4.40	18	EDARLIRN	0.180	237
LTAG13.EDAF14	5.20	18	EDARLIRN	0.180	237
EGUN14.EDAF14	2.09	7	EDARLIRN	0.190	251
LETO11.EDAF11	2.60	18	EDARLIRN	0.180	262
EDAF1 .LIRN1	2.10	18	EDARLIRN	0.360	264
EDAF14.LIRN14	2.04	50	EDARLIRN	0.370	266
EDAF7 .LIRN7	2.10	18	EDARLIRN	0.550	266
EDAR10.EDAF10	0.10	18	EDARLIRN	0.180	292
EDAR2 .LLBG3	4.17	50	EDARLIRN	0.180	293
LLBG3 .EDAR4	5.24	50	EDARLIRN	0.180	293
EGUN16.EDAR16	1.40	18	EDAROEDR	0.590	59
EDAF3 .OEDR4	6.64	71	EDAROEDR	1.700	224
EDAF10.LETO11	2.60	18	EDAROEDR	0.850	230
EDAR20.EGUN21	1.50	18	EDAROEDR	0.850	230
EGUN21.EDAF21	1.50	18	EDAROEDR	0.850	230
EDAR8 .EGUN8	1.50	18	EDAROEDR	1.690	230
EGUN8 .EDAF9	1.50	18	EDAROEDR	1.690	230
EDAF17.OKBK18	4.90	18	EDAROEDR	1.690	235
OKBK18.OEDR18	0.60	18	EDAROEDR	1.690	235
EDAF12.LTAG13	4.40	18	EDAROEDR	0.850	237
LTAG13.EDAF14	5.20	18	EDAROEDR	0.850	237
EDAR15.EGUN16	2.09	7	EDAROEDR	0.590	249
LETO11.EDAF11	2.60	18	EDAROEDR	0.850	262
EDAF14.OEDR15	7.30	18	EDAROEDR	0.850	271
EDAF9 .OEDR10	7.30	18	EDAROEDR	1.690	271
EDAR2 .EDAF2	0.10	18	EDAROEDR	0.850	292
EDAR10.EDAF10	0.10	18	EDAROEDR	0.850	292
EDAR17.EDAF17	0.10	18	EDAROEDR	1.690	292
EDAF10.KCHS11	10.60	18	EGUNKNGU	0.780	59
EGUN16.EDAR16	1.40	18	EGUNKNGU	0.780	59
EDAR16.KDOV17	9.50	18	EGUNKNGU	0.780	200
EGUN4 .KCHS6	9.02	25	EGUNKNGU	2.340	202
KCHS11.KNGU11	1.10	18	EGUNKNGU	0.780	216
EGUN8 .EDAF9	1.50	18	EGUNKNGU	0.780	230
EGUN14.EDAR14	1.40	18	EGUNKNGU	0.780	231
EGUN10.EDAR11	1.95	7	EGUNKNGU	0.780	249
EDAR14.KDOV15	8.74	30	EGUNKNGU	1.560	252
KDOV17.KNGU17	0.80	18	EGUNKNGU	2.340	255
KCHS7 .KNGU7	1.05	75	EGUNKNGU	2.340	259
EGUN16.EDAR16	1.40	18	EGUNLTAG	1.680	59
EDAR16.EDAF16	0.10	18	EGUNLTAG	1.960	59

EGUN21.EDAF21	1.50	18	EGUNLTAG	1.680	230
EGUN8 .EDAF9	1.50	18	EGUNLTAG	1.680	230
EGUN14.EDAR14	1.40	18	EGUNLTAG	1.680	231
EDAR17.LIPA17	1.50	18	EGUNLTAG	1.400	231
EDAF4 .LTAG4	4.40	18	EGUNLTAG	5.040	237
EGUN10.EDAR11	1.95	7	EGUNLTAG	3.360	249
LIPA17.LGIR17	3.20	7	EGUNLTAG	3.360	251
LGIR17.LCRA18	1.95	7	EGUNLTAG	3.360	251
LCRA18.LTAG18	2.50	7	EGUNLTAG	3.360	251
EGUN4 .EDAF4	2.09	7	EGUNLTAG	3.360	251
EDAR12.LTAG12	4.14	30	EGUNLTAG	3.360	252
EDAF9 .EGUN9	1.50	18	EGUNLTAG	1.680	262
EDAF16.LETO16	2.60	18	EGUNLTAG	1.960	262
LETO16.LIPA17	2.20	18	EGUNLTAG	1.960	262
KCHS13.KTIK13	2.80	18	KCHSEDAF	0.290	59
KTIK15.KXXX15	0.00	71	KCHSEDAF	0.290	137
KXXX17.KTIK17	0.00	25	KCHSEDAF	0.290	137
KTIK17.EDAF19	11.35	25	KCHSEDAF	0.290	137
KCHS1 .KNGU1	1.02	25	KCHSEDAF	0.650	202
KNGU1 .BIKF2	5.49	25	KCHSEDAF	0.650	202
BIKF4 .EGUN4	2.98	25	KCHSEDAF	0.650	202
EGUN4 .EDAF4	2.09	7	KCHSEDAF	0.650	251
KCHS7 .KNGU7	1.05	75	KCHSEDAF	0.020	259
KNGU8 .LERT9	7.47	50	KCHSEDAF	0.020	259
KCHS10.KNGU11	1.07	50	KCHSEDAF	0.240	260
KNGU11.LERT12	7.47	50	KCHSEDAF	0.240	260
LERT14.LIRN15	2.72	50	KCHSEDAF	0.260	260
LERT10.LIRN11	2.60	25	KCHSEDAF	0.020	260
LIRN11.LERT12	2.79	25	KCHSEDAF	0.020	260
KCHS4 .KNGU4	1.10	18	KCHSEDAF	0.040	265
KNGU4 .LERT5	7.70	18	KCHSEDAF	0.040	265
LERT7 .LIRN8	2.80	18	KCHSEDAF	0.040	265
LIRN16.EDAF16	2.13	50	KCHSEDAF	0.260	266
LIRN9 .EDAF9	2.20	18	KCHSEDAF	0.040	266
KDOV14.EDAF15	7.68	146	KDOVLGIR	0.420	56(C5), 180(B747), 269(DC8)
KDOV21.KTIK21	3.13	75	KDOVLGIR	0.480	56
KTIK1 .KDOV2	2.90	18	KDOVLGIR	0.480	59
KDOV10.EDAF11	7.95	50	KDOVLGIR	0.360	180
KDOV17.EDAR18	7.52	141	KDOVLGIR	0.490	181(B747), 181(DC10), 252(KC10)
LETO5 .LIPA5	2.20	18	KDOVLGIR	0.850	230
EDAF12.LTAG13	4.40	18	KDOVLGIR	0.360	237
LTAG13.EDAF14	5.20	18	KDOVLGIR	0.360	237
KDOV4 .LETO5	7.18	96	KDOVLGIR	0.850	241(B747), 241(DC8)
EDAF14.LIPA15	2.64	7	KDOVLGIR	0.360	251
LIPA17.LGIR17	3.20	7	KDOVLGIR	0.780	251

LTAG20.LCRA20	1.81	7	KDOVLGIR	0.490	251
LCRA21.LGIR21	1.95	7	KDOVLGIR	0.490	251
LIPA7 .LGIR7	3.20	7	KDOVLGIR	0.850	251
EDAR19.LTAG19	4.14	30	KDOVLGIR	0.490	252
EDAF16.LETO16	2.60	18	KDOVLGIR	0.420	262
LETO16.LIPA17	2.20	18	KDOVLGIR	0.420	262
KDOV14.EDAF15	7.68	146	KDOVLIPA	3.450	56(C5), 180(B747), 269(DC8)
KDOV21.KTIK21	3.13	75	KDOVLIPA	9.670	56
KTIK1 .KDOV2	2.90	18	KDOVLIPA	9.670	59
[illegible]13 .EGUN14	7.10	18	KDOVLIPA	4.950	59
[illegible].EDAF11	7.95	50	KDOVLIPA	0.150	180
KDOV17.EDAR18	7.52	141	KDOVLIPA	9.860	181(B747), 181(DC10), 252(KC10)
KDOV7 .EDAR8	8.20	18	KDOVLIPA	1.350	200
EDAR20.EGUN21	1.50	18	KDOVLIPA	9.860	230
EGUN21.EDAF21	1.50	18	KDOVLIPA	9.860	230
LETO5 .LIPA5	2.20	18	KDOVLIPA	15.820	230
EDAR8 .LIPA8	1.50	18	KDOVLIPA	9.170	231
EDAF12.LTAG13	4.40	18	KDOVLIPA	0.150	237
LTAG13.EDAF14	5.20	18	KDOVLIPA	0.150	237
KDOV4 .LETO5	7.18	96	KDOVLIPA	15.820	241(B747), 241(DC8)
EDAR11.LIRP11	4.03	7	KDOVLIPA	7.000	249
LIRP12.LIPA12	1.11	7	KDOVLIPA	7.000	249
EGUN14.EDAF14	2.09	7	KDOVLIPA	4.950	251
EDAF14.LIPA15	2.64	7	KDOVLIPA	5.100	251
EDAF4 .LIPA5	2.64	7	KDOVLIPA	2.040	251
KDOV10.EDAR11	7.54	30	KDOVLIPA	7.000	252
EDAF16.LETO16	2.60	18	KDOVLIPA	3.450	262
LETO16.LIPA17	2.20	18	KDOVLIPA	3.450	262
EDAF2 .EDAR2	0.10	18	KDOVLIPA	7.820	292
EDAR2 .LLBG3	4.17	50	KDOVLIPA	7.820	293
LLBG3 .EDAR4	5.24	50	KDOVLIPA	7.820	293
KDOV14.EDAF15	7.68	146	KDOVOEDR	6.580	56(C5), 180(B747), 269(DC8)
KDOV4 .EGUN5	7.10	18	KDOVOEDR	1.090	59
KDOV13.EGUN14	7.10	18	KDOVOEDR	1.860	59
KDOV10.EDAF11	7.95	50	KDOVOEDR	7.170	180
KDOV17.EDAR18	7.52	141	KDOVOEDR	9.900	181(B747), 181(DC10), 252(KC10)
KDOV7 .EDAR8	8.20	18	KDOVOEDR	0.180	200
KDOV1 .EDAF1	7.46	71	KDOVOEDR	15.970	224
EDAF3 .OEDR4	6.64	71	KDOVOEDR	15.970	224
EDAR8 .EGUN8	1.50	18	KDOVOEDR	0.180	230
EGUN8 .EDAF9	1.50	18	KDOVOEDR	1.270	230

EDAF17.OKBK18	4.90	18	KDOVOEDR	6.580	235
OKBK18.OEDR18	0.60	18	KDOVOEDR	6.580	235
EDAF12.LTAG13	4.40	18	KDOVOEDR	7.170	237
LTAG13.EDAF14	5.20	18	KDOVOEDR	7.170	237
EGUN14.EDAF14	2.09	7	KDOVOEDR	1.860	251
EDAR18.EGUN18	1.50	18	KDOVOEDR	9.900	262
EGUN18.EDAF18	1.50	18	KDOVOEDR	9.900	262
EDAF14.OEDR15	7.30	18	KDOVOEDR	9.030	271
EDAF19.OEDR20	7.30	18	KDOVOEDR	9.900	271
EDAF9 .OEDR10	7.30	18	KDOVOEDR	1.270	271
KDOV7 .EDAR8	8.20	18	KNGULIPA	4.750	200
KCHS1 .KNGU1	1.02	25	KNGULIPA	1.800	202
KNGU1 .BIKF2	5.49	25	KNGULIPA	2.990	202
BIKF4 .EGUN4	2.98	25	KNGULIPA	2.990	202
KCHS20.KNGU20	1.10	36	KNGULIPA	4.750	216(C141), 2[illegible](C141)
EDAR8 .LIPA8	1.50	18	KNGULIPA	4.750	231
LETO4 .KDOV5	8.44	50	KNGULIPA	4.750	241
EGUN4 .EDAF4	2.09	7	KNGULIPA	2.990	251
EDAF4 .LIPA5	2.64	7	KNGULIPA	2.990	251
KNGU8 .LERT9	7.47	50	KNGULIPA	0.270	259
KNGU11.LERT12	7.47	50	KNGULIPA	1.940	260
LERT14.LIRN15	2.72	50	KNGULIPA	2.210	260
KNGU19.KCHS20	1.07	50	KNGULIPA	6.550	260
LERT10.LIRN11	2.60	25	KNGULIPA	0.270	260
LIRN11.LERT12	2.79	25	KNGULIPA	0.270	260
EDAR10.LIPA10	1.50	18	KNGULIPA	0.550	262
EDAF16.LETO16	2.60	18	KNGULIPA	2.210	262
LETO16.LIPA17	2.20	18	KNGULIPA	2.210	262
KNGU4 .LERT5	7.70	18	KNGULIPA	0.550	265
LERT7 .LIRN8	2.80	18	KNGULIPA	0.550	265
LIRN16.EDAF16	2.13	50	KNGULIPA	2.210	266
LIRN9 .EDAF9	2.20	18	KNGULIPA	0.550	266
EDAF10.EDAR10	0.10	18	KNGULIPA	0.550	292
KNGU20.LETO21	8.10	18	KNGULIPA	4.750	294
KDOV21.KTIK21	3.13	75	KTIKLIPA	0.820	56
KTIK1 .KDOV2	2.90	18	KTIKLIPA	1.330	59
KTIK19.KDOV20	2.90	18	KTIKLIPA	0.820	59
KTIK11.EDAF12	11.10	71	KTIKLIPA	0.540	137
KTIK15.KXXX15	0.00	71	KTIKLIPA	0.850	137
KXXX17.KTIK17	0.00	25	KTIKLIPA	0.850	137
KTIK17.EDAF19	11.35	25	KTIKLIPA	1.670	137
KTIK8 .KWRI8	2.90	18	KTIKLIPA	0.400	225
KWRI10.LPLA11	5.40	36	KTIKLIPA	0.400	225(C141), 270(C141)
LPLA11.EDAF12	4.40	18	KTIKLIPA	0.400	225
LETO5 .LIPA5	2.20	18	KTIKLIPA	1.330	230
EDAF12.LTAG13	4.40	18	KTIKLIPA	0.940	237
LTAG13.EDAF14	5.20	18	KTIKLIPA	0.940	237

KDOV4 .LETO5	7.18	96	KTIKLIPA	1.330	241(B747), 241(DC8)
EDAF14.LIPA15	2.64	7	KTIKLIPA	0.940	251
EDAF4 .LIPA5	2.64	7	KTIKLIPA	1.670	251
KDOV21.KTIK21	3.13	75	KTIKLTAG	1.330	56
KTIK15.KDOV15	2.81	50	KTIKLTAG	1.380	58
KTIK1 .KDOV2	2.90	18	KTIKLTAG	2.160	59
KDOV4 .EGUN5	7.10	18	KTIKLTAG	2.160	59
KTIK19.KDOV20	2.90	18	KTIKLTAG	1.330	59
KTIK11.EDAF12	11.10	71	KTIKLTAG	0.880	137
KTIK17.EDAF19	11.35	25	KTIKLTAG	1.330	137
KDOV17.EDAR18	7.52	141	KTIKLTAG	1.380	181(B747), 181(DC10), 252(KC10)
KTIK8 .KWRI8	2.90	[illegible]	[illegible]IKLTAG	0.640	225
KWRI10.LPLA11	5.40	[illegible]	[illegible]KLTAG	0.640	225(C141), 270(C141)
EGUN5 .EDAR5	1.40	18	KTIKLTAG	2.160	231
EDAF20.LTAG21	4.40	18	KTIKLTAG	1.330	237
EDAR12.LTAG12	4.14	30	KTIKLTAG	0.640	252
EDAR19.LTAG19	4.14	30	KTIKLTAG	1.380	252
EDAR5 .LTAG5	4.14	30	KTIKLTAG	2.160	252
EDAF12.LTAG12	4.40	18	KTIKLTAG	0.880	262
LPLA11.EDAR12	4.60	18	KTIKLTAG	0.640	270
KTIK1 .KDOV2	2.90	18	KTIKOEDR	0.940	59
KDOV4 .EGUN5	7.10	18	KTIKOEDR	0.940	59
KTIK19.KDOV20	2.90	18	KTIKOEDR	1.510	59
KTIK11.EDAF12	11.10	71	KTIKOEDR	0.980	137
KTIK17.EDAF19	11.35	25	KTIKOEDR	3.070	137
KDOV1 .EDAF1	7.46	71	KTIKOEDR	1.510	224
EDAF3 .OEDR4	6.64	71	KTIKOEDR	1.510	224
KTIK8 .KWRI8	2.90	18	KTIKOEDR	0.730	225
KWRI10.LPLA11	5.40	36	KTIKOEDR	0.730	225(C141), 270(C141)
LPLA11.EDAF12	4.40	18	KTIKOEDR	0.730	225
EGUN8 .EDAF9	1.50	18	KTIKOEDR	0.940	230
EDAF12.LTAG13	4.40	18	KTIKOEDR	1.560	237
LTAG13.EDAF14	5.20	18	KTIKOEDR	1.560	237
EDAF14.OEDR15	7.30	18	KTIKOEDR	1.710	271
EDAF19.OEDR20	7.30	18	KTIKOEDR	3.070	271
EDAF9 .OEDR10	7.30	18	KTIKOEDR	0.940	271
KDOV14.EDAF15	7.68	146	KTIKOERY	0.530	56(C5), 180(B747), 269(DC8)
KTIK1 .KDOV2	2.90	18	KTIKOERY	0.500	59
KDOV4 .EGUN5	7.10	18	KTIKOERY	0.500	59
KTIK10.KDOV11	2.90	18	KTIKOERY	0.530	59
KTIK19.KDOV20	2.90	18	KTIKOERY	0.800	59
KTIK17.EDAF19	11.35	25	KTIKOERY	1.650	137
KDOV1 .EDAF1	7.46	71	KTIKOERY	0.800	224

KTIK8 .KWRI8	2.90	18	KTIKOERY	0.390	225
KWRI10.LPLA11	5.40	36	KTIKOERY	0.390	225(C141), 270(C141)
LPLA11.EDAF12	4.40	18	KTIKOERY	0.390	225
EDAF10.LETO11	2.60	18	KTIKOERY	0.500	230
EGUN15.EDAF15	1.50	18	KTIKOERY	0.890	230
EGUN8 .EDAF9	1.50	18	KTIKOERY	0.500	230
OEDR20.OERY20	1.00	18	KTIKOERY	1.650	235
EDAF4 .OKBK5	4.90	18	KTIKOERY	0.800	235
OKBK5 .OEDR5	0.60	18	KTIKOERY	0.800	235
OEDR7 .OERY7	1.00	18	KTIKOERY	0.800	235
EDAF12.LTAG13	4.40	18	KTIKOERY	0.890	237
LTAG13.EDAF14	5.20	18	KTIKOERY	0.890	237
EDAF14.EGUN15	2.09	7	KTIKOERY	0.890	251
LETO11.EDAF11	2.60	18	KTIKOERY	0.500	262
EDAF15.OERY16	6.42	25	KTIKOERY	1.420	269
EDAF19.OEDR20	7.30	18	KTIKOERY	1.650	271
EGUN16.EDAR16	1.40	18	LETOKDOV	6.410	59
EDAF13.KDOV14	9.01	71	LETOKDOV	8.190	137
EDAF17.KDOV18	9.01	71	LETOKDOV	9.960	180
EDAR16.KDOV17	9.50	18	LETOKDOV	6.410	200
LETO4 .KDOV5	8.44	50	LETOKDOV	24.560	241
LETO7 .KDOV8	8.00	96	LETOKDOV	8.190	241(B747), 241(DC8)
LETO15.EDAR15	3.47	7	LETOKDOV	6.410	249
EDAR15.EGUN16	2.09	7	LETOKDOV	6.410	249
LETO11.EDAF11	2.60	18	LETOKDOV	8.190	262
LETO16.LIPA17	2.20	18	LETOKDOV	9.960	262
LIPA17.EDAR17	1.80	18	LETOKDOV	9.960	262
EDAR17.EDAF17	0.10	18	LETOKDOV	9.960	292
KDOV21.KTIK21	3.13	75	LETOKTIK	1.540	56
EDAF10.KCHS11	10.60	18	LETOKTIK	3.080	59
KCHS13.KTIK13	2.80	18	LETOKTIK	3.080	59
EDAF13.KDOV14	9.01	71	LETOKTIK	0.770	137
KDOV14.KTIK15	3.00	71	LETOKTIK	0.770	137
EDAR20.KDOV21	8.69	111	LETOKTIK	1.540	181(B747), 181(DC10)
KDOV7 .EDAR8	8.20	18	LETOKTIK	2.310	200
LETO17.LIPA18	2.20	18	LETOKTIK	1.540	230
LIPA20.EDAR20	1.80	18	LETOKTIK	1.540	230
EDAR8 .EGUN8	1.50	18	LETOKTIK	2.310	230
EGUN8 .EDAF9	1.50	18	LETOKTIK	2.310	230
LETO9 .EDAF9	2.60	18	LETOKTIK	0.770	231
LETO4 .KDOV5	8.44	50	LETOKTIK	2.310	241
LETO11.EDAF11	2.60	18	LETOKTIK	0.770	262
KDOV21.KTIK21	3.13	75	LETOKWRI	2.320	56
EDAR20.KDOV21	8.69	111	LETOKWRI	2.320	181(B747), 181(DC10)
KDOV7 .EDAR8	8.20	18	LETOKWRI	3.480	200
KTIK8 .KWRI8	2.90	18	LETOKWRI	2.320	225

EDAF14.KWRI15	9.70	18	LETOKWRI	2.940	225
EDAF10.LETO11	2.60	18	LETOKWRI	1.780	230
LETO17.LIPA18	2.20	18	LETOKWRI	2.320	230
LIPA20.EDAR20	1.80	18	LETOKWRI	2.320	230
EDAR8 .EGUN8	1.50	18	LETOKWRI	3.480	230
EGUN8 .EDAF9	1.50	18	LETOKWRI	3.480	230
LETO9 .EDAF9	2.60	18	LETOKWRI	1.160	231
EDAF12.LTAG13	4.40	18	LETOKWRI	2.940	237
LTAG13.EDAF14	5.20	18	LETOKWRI	2.940	237
LETO4 .KDOV5	8.44	50	LETOKWRI	3.480	241
EGUN10.EDAR11	1.95	7	LETOKWRI	2.860	249
EDAF9 .EGUN9	1.50	18	LETOKWRI	2.860	262
LETO11.EDAF11	2.60	18	LETOKWRI	2.940	262
EDAR14.LPLA15	4.60	18	LETOKWRI	2.860	270
LPLA15.KWRI16	6.40	18	LETOKWRI	2.860	270
EDAF10.KCHS11	10.60	18	LETOLERT	0.600	59
KDOV6 .EDAF7	7.46	71	LETOLERT	1.790	180
EDAR16.KDOV17	9.50	18	LETOLERT	0.590	200
KCHS11.KNGU11	1.10	18	LETOLERT	0.600	216
LETO17.LIPA18	2.20	18	LETOLERT	0.600	230
LIPA20.EDAR20	1.80	18	LETOLERT	0.600	230
EDAR20.EGUN21	1.50	18	LETOLERT	0.600	230
EGUN21.EDAF21	1.50	18	LETOLERT	0.600	230
LETO9 .EDAF9	2.60	18	LETOLERT	0.600	231
EDAF12.LTAG13	4.40	18	LETOLERT	0.600	237
LTAG13.EDAF14	5.20	18	LETOLERT	0.600	237
LETO4 .KDOV5	8.44	50	LETOLERT	1.790	241
LETO15.EDAR15	3.47	7	LETOLERT	0.590	249
KDOV17.KNGU17	0.80	18	LETOLERT	0.590	255
KNGU18.LERT18	7.70	18	LETOLERT	0.590	255
KNGU11.LERT12	7.47	50	LETOLERT	0.600	260
LIRN15.LERT16	2.91	50	LETOLERT	0.600	260
LIRN11.LERT12	2.79	25	LETOLERT	1.790	260
LETO11.EDAF11	2.60	18	LETOLERT	0.600	262
EDAF1 .LIRN1	2.10	18	LETOLERT	0.600	264
LIRN1 .LICZ1	1.00	18	LETOLERT	0.600	264
LICZ2 .LERT2	3.20	18	LETOLERT	0.600	264
EDAF14.LIRN14	2.04	50	LETOLERT	0.600	266
LIRN14.LICZ15	0.97	50	LETOLERT	0.600	266
LICZ15.LIRN15	0.97	50	LETOLERT	0.600	266
EDAF7 .LIRN7	2.10	18	LETOLERT	1.790	266
KDOV6 .EDAF7	7.46	71	LETOLIRN	2.650	180
EDAF10.LETO11	2.60	18	LETOLIRN	0.880	230
LETO17.LIPA18	2.20	18	LETOLIRN	1.770	230
LIPA20.EDAR20	1.80	18	LETOLIRN	1.770	230
EDAR20.EGUN21	1.50	18	LETOLIRN	1.770	230
EGUN21.EDAF21	1.50	18	LETOLIRN	1.770	230
LETO9 .EDAF9	2.60	18	LETOLIRN	0.880	231
EDAF12.LTAG13	4.40	18	LETOLIRN	1.760	237
LTAG13.EDAF14	5.20	18	LETOLIRN	1.760	237

LETO4 .KDOV5	8.44	50	LETOLIRN	2.650	241
LETO11.EDAF11	2.60	18	LETOLIRN	1.760	262
EDAF1 .LIRN1	2.10	18	LETOLIRN	1.770	264
EDAF14.LIRN14	2.04	50	LETOLIRN	1.760	266
EDAF7 .LIRN7	2.10	18	LETOLIRN	2.650	266

Appendix P: GAMS Program for Example Problem (Version 1)

This appendix contains the GAMS Program for the first version of the example problem in Chapter IV.

```
SET K   commodities (cargo)
 /AB, AC, BA, BC, CA, CB/;

SET I   airbases(AB)-time periods
 /A1 * A7, B1 * B7, C1 * C7/;

ALIAS (I,J);

ALIAS (I,IP);

SET DIK(I,K)  dynamic set for IK;
DIK(I,K) = yes;

SET E1(I,J,K)  arcs for cargo staying at AB
 /(A1.A2, A2.A3, A3.A4, A4.A5, A5.A6, A6.A7, A7.A1,
  B1.B2, B2.B3, B3.B4, B4.B5, B5.B6, B6.B7, B7.B1,
  C1.C2, C2.C3, C3.C4, C4.C5, C5.C6, C6.C7, C7.C1).
  (AB, AC, BA, BC, CA, CB)/;

SET Et(I,J,K)  dynamic set for E1;
Et(I,J,K) = no;
Et(E1) = yes;

SET E2(I,J,K)  arcs representing a-c with cargo
 /(C7.A1, A1.B2, B2.C3, C3.A4, A4.B5, B5.C6,
  B1.C2, C2.A3, A3.C4, C4.B5,
  B4.C5, C5.B6).
  (AB, AC, BA, BC, CA, CB)/;

SET Es(I,J,K)  dynamic set for E2;
Es(I,J,K) = no;
Es(E2) = yes;

SET E(I,J,K)  set of all arcs (Et and Es);
E(I,J,K) = Et(I,J,K) + Es(I,J,K);

SET E3(I,J)  arcs representing aircraft
 /C7.A1, A1.B2, B2.C3, C3.A4, A4.B5, B5.C6,
  B1.C2, C2.A3, A3.C4, C4.B5,
  B4.C5, C5.B6/

SET SIKN(I,K)  supply nodes for all cargo
 /A1.AB, A2.AB, A3.AB, A4.AB, A5.AB, A6.AB, A7.AB,
  A1.AC, A2.AC, A3.AC, A4.AC, A5.AC, A6.AC, A7.AC,
```

```
  B1.BA, B2.BA, B3.BA, B4.BA, B5.BA, B6.BA, B7.BA,
  B1.BC, B2.BC, B3.BC, B4.BC, B5.BC, B6.BC, B7.BC,
  C1.CA, C2.CA, C3.CA, C4.CA, C5.CA, C6.CA, C7.CA,
  C1.CB, C2.CB, C3.CB, C4.CB, C5.CB, C6.CB, C7.CB/;


SET SUPNODE(I,K)  dynamic set for SIKN;
SUPNODE(I,K) = no;
SUPNODE(SIKN) = yes;

SET DIKN(I,K)  airbase demand nodes for all cargo
 /A1.BA, A2.BA, A3.BA, A4.BA, A5.BA, A6.BA, A7.BA,
  A1.CA, A2.CA, A3.CA, A4.CA, A5.CA, A6.CA, A7.CA,
  B1.AB, B2.AB, B3.AB, B4.AB, B5.AB, B6.AB, B7.AB,
  B1.CB, B2.CB, B3.CB, B4.CB, B5.CB, B6.CB, B7.CB,
  C1.AC, C2.AC, C3.AC, C4.AC, C5.AC, C6.AC, C7.AC,
  C1.BC, C2.BC, C3.BC, C4.BC, C5.BC, C6.BC, C7.BC/;

SET DMDNODE(I,K)  dynamic set for DIKN;
DMDNODE(I,K) = no;
DMDNODE(DIKN) = yes;

SET ZIKN(I,K)  neither demand nor supply nodes;
ZIKN(I,K) = DIK(I,K) - SUPNODE(I,K) - DMDNODE(I,K);

SET ZN(I)  airbases that serve as zero balance nodes
 /A1 * A7, B1 * B7, C1 * C7/;

PARAMETER C(I,J,K)  delay;

C(I,J,K) = 0;

C(I,J,K)$Et(I,J,K) = 1;

C(I,J,K)$Es(I,J,K) = 1;

PARAMETER S(I,K) the supply at node SIKN
 /A1.AB 2, A2.AB 5, A3.AB 6, A4.AB 12, A5.AB 6, A6.AB 5,
A7.AB 2,
  A1.AC 1, A2.AC 2, A3.AC 2, A4.AC 5, A5.AC 3, A6.AC 2,
A7.AC 1,
  B1.BA 1, B2.BA 2, B3.BA 2. B4.BA 4, B5.BA 2, B6.BA 2,
B7.BA 1,
  B1.BC 2, B2.BC 3, B3.BC 5, B4.BC 8, B5.BC 5, B6.BC 3,
B7.BC 2,
  C1.CA 0, C2.CA 1, C3.CA 2, C4.CA 3, C5.CA 2, C6.CA 0,
C7.CA 0,
  C1.CB 2, C2.CB 3, C3.CB 5, C4.CB 8, C5.CB 5, C6.CB 3,
C7.CB 2/;
```

```
PARAMETER CAP(I,J)  aircraft capacity
 /C7.A1 18, A1.B2 18, B2.C3 18, C3.A4 18, A4.B5 18, B5.C6
18,
  B1.C2 25, C2.A3 25, A3.C4 25, C4.B5 25,
  B4.C5 30, C5.B6 30/;

VARIABLE
Z  total delay

POSITIVE VARIABLES
X(I,J,K) shipment quantity
SUP(K)  total supply for each cargo K
DEL(K)  total amount delivered for each cargo
UNDEL(K)  total amount not delivered for each cargo;

EQUATIONS
DELAY  objective function
SUMS(K)  total supply for each cargo K
SUPPLY(IP,K)  conservation of flow for supply nodes
DEMAND(IP,K)  conservation of flow for demand nodes
DELIVER(K)  total amount delivered for each cargo
UNDELIVER(K)  total amount not delivered for each cargo
BAL(IP,K)  conservation of flow for ZIKN nodes
UB(I,J)  upper bound capacity constraint for aircraft;

DELAY .. Z =E= SUM((I,J,K)$E(I,J,K), C(I,J,K)*X(I,J,K));

SUMS(K) .. SUP(K) =E= SUM(I,S(I,K));

SUPPLY(IP,K)$SIKN(IP,K) .. SUM(J, X(IP,J,K)$E(IP,J,K)) -
                           SUM(I, X(I,IP,K)$E(I,IP,K))
                           =E= S(IP,K);

DEMAND(IP,K)$DIKN(IP,K) .. SUM(J, X(IP,J,K)$E(IP,J,K)) -
                           SUM(I, X(I,IP,K)$E(I,IP,K))
                           =G= -SUP(K);

DELIVER(K) .. DEL(K) =E= SUM((I,IP)$E3(I,IP),
X(I,IP,K)$DIKN(IP,K));

UNDELIVER(K) .. UNDEL(K) =E= SUP(K) - DEL(K);

BAL(IP,K)$ZIKN(IP,K) ..  SUM(J, X(IP,J,K)$E(IP,J,K)) -
                         SUM(I, X(I,IP,K)$E(I,IP,K))
                         =E= 0;

UB( E3(I,J) ) ..  SUM(K, X(I,J,K)) =L= CAP(E3);

MODEL MMCF /ALL/;
```

```
OPTION ITERLIM = 10000, RESLIM = 10000;
OPTION LIMROW = 0, LIMCOL = 0;

SOLVE MMCF USING LP MINIMIZING Z;
```

Appendix Q: Results for Example Problem (Version 1)

This appendix contains a portion of the results from the GAMS program for the first version of the example problem in Chapter IV.

```
**** OBJECTIVE VALUE                 310.0000

---- EQU UB   UPPER BOUND CAPACITY CONSTRAINT FOR AIRCRAFT

         LOWER     LEVEL     UPPER    MARGINAL

A1.B2     -INF     18.000    18.000    -4.000
A3.C4     -INF     22.000    25.000      .
A4.B5     -INF     18.000    18.000    -1.000
B1.C2     -INF     11.000    25.000      .
B2.C3     -INF      5.000    18.000      .
B4.C5     -INF     19.000    30.000      .
B5.C6     -INF     12.000    18.000      .
C2.A3     -INF      5.000    25.000      .
C3.A4     -INF      5.000    18.000      .
C4.B5     -INF     25.000    25.000    -1.000
C5.B6     -INF     10.000    30.000      .
C7.A1     -INF     16.000    18.000      .
```

Appendix R: Results for Example Problem (Version 2)

This appendix contains a portion of the results from the GAMS program for the second version of the example problem in Chapter IV.

```
**** OBJECTIVE VALUE                294.0000

---- EQU UB    UPPER BOUND CAPACITY CONSTRAINT FOR AIRCRAFT

            LOWER      LEVEL      UPPER     MARGINAL

A1.B2        -INF      25.000     25.000     -1.000
A3.C4        -INF      14.000     30.000       .
A4.B5        -INF      25.000     25.000      EPS
B1.C2        -INF      11.000     30.000       .
B2.C3        -INF      10.000     25.000       .
B4.C5        -INF      13.000     18.000       .
B5.C6        -INF      21.000     25.000       .
C2.A3        -INF      10.000     30.000       .
C3.A4        -INF       4.000     25.000       .
C4.B5        -INF      24.000     30.000       .
C5.B6        -INF       5.000     18.000       .
C7.A1        -INF      18.000     25.000       .
```

Appendix S: Results for Example Problem (Version 3)

This appendix contains a portion of the results from the GAMS program for the third version of the example problem in Chapter IV.

```
**** OBJECTIVE VALUE                292.0000

---- EQU UB   UPPER BOUND CAPACITY CONSTRAINT FOR AIRCRAFT

          LOWER      LEVEL      UPPER     MARGINAL

A1.B2      -INF      27.000     30.000       .
A3.C4      -INF      20.000     25.000       .
A4.B5      -INF      22.000     30.000       .
B1.C2      -INF      11.000     25.000       .
B2.C3      -INF      12.000     30.000       .
B4.C5      -INF      18.000     18.000      EPS
B5.C6      -INF      14.000     30.000       .
C2.A3      -INF      10.000     25.000       .
C3.A4      -INF       9.000     30.000       .
C4.B5      -INF      25.000     25.000      EPS
C5.B6      -INF       5.000     18.000       .
C7.A1      -INF      18.000     30.000       .
```

Appendix T: Results for Example Problem (Version 3)

This appendix contains a portion of the results from the GAMS program for the third version of the example problem in Chapter IV.

```
**** OBJECTIVE VALUE                292.0000

     EQU SUPPLY      CONSERVATION OF FLOW FOR SUPPLY NODES

             LOWER       LEVEL       UPPER      MARGINAL

A2.AC        2.000       2.000       2.000       2.000
A3.AB        6.000       6.000       6.000       2.000
A3.AC        2.000       2.000       2.000       1.000
A4.AB       12.000      12.000      12.000       1.000
A4.AC        5.000       5.000       5.000       2.000
A5.AB        6.000       6.000       6.000       4.000
A5.AC        3.000       3.000       3.000       5.000
A6.AB        5.000       5.000       5.000       3.000
A6.AC        2.000       2.000       2.000       4.000
A7.AB        2.000       2.000       2.000       2.000
A7.AC        1.000       1.000       1.000       3.000
B1.BA        1.000       1.000       1.000       2.000
B1.BC        2.000       2.000       2.000       1.000
B2.BA        2.000       2.000       2.000       2.000
B2.BC        3.000       3.000       3.000       1.000
B3.BA        2.000       2.000       2.000       5.000
B3.BC        5.000       5.000       5.000       2.000
B4.BA        4.000       4.000       4.000       4.000
B4.BC        8.000       8.000       8.000       1.000
B5.BA        2.000       2.000       2.000       3.000
B5.BC        5.000       5.000       5.000       1.000
B6.BA        2.000       2.000       2.000       4.000
B6.BC        3.000       3.000       3.000       3.000
B7.BA        1.000       1.000       1.000       3.000
B7.BC        2.000       2.000       2.000       2.000
C1.CA          .           .           .         2.000
C1.CB        2.000       2.000       2.000       4.000
C2.CA        1.000       1.000       1.000       1.000
C2.CB        3.000       3.000       3.000       3.000
C3.CA        2.000       2.000       2.000       1.000
C3.CB        5.000       5.000       5.000       2.000
C4.CA        3.000       3.000       3.000       4.000
C4.CB        8.000       8.000       8.000       1.000
C5.CA        2.000       2.000       2.000       3.000
C5.CB        5.000       5.000       5.000       1.000
C6.CA          .           .           .         2.000
C6.CB        3.000       3.000       3.000       3.000
```

C7.CA	.	.	.	1.000
C7 [illegible]B	2.000	2.000	2.000	2.000

Appendix U: Results for Example Problem (Version 4)

This appendix contains a portion of the results from the GAMS program for the fourth version of the example problem in Chapter IV.

```
**** OBJECTIVE VALUE              273.0000

     EQU SUPPLY      CONSERVATION OF FLOW FOR SUPPLY NODES

             LOWER       LEVEL       UPPER     MARGINAL

A2.AC        2.000       2.000       2.000       4.000
A3.AB        6.000       6.000       6.000       3.000
A3.AC        2.000       2.000       2.000       3.000
A4.AB       12.000      12.000      12.000       2.000
A4.AC        5.000       5.000       5.000       2.000
A5.AB        6.000       6.000       6.000       2.000
A5.AC        3.000       3.000       3.000       1.000
A6.AB        5.000       5.000       5.000       3.000
A6.AC        2.000       2.000       2.000       4.000
A7.AB        2.000       2.000       2.000       2.000
A7.AC        1.000       1.000       1.000       3.000
B1.BA        1.000       1.000       1.000       3.000
B1.BC        2.000       2.000       2.000       3.000
B2.BA        2.000       2.000       2.000       2.000
B2.BC        3.000       3.000       3.000       2.000
B3.BA        2.000       2.000       2.000       2.000
B3.BC        5.000       5.000       5.000       2.000
B4.BA        4.000       4.000       4.000       4.000
B4.BC        8.000       8.000       8.000       2.000
B5.BA        2.000       2.000       2.000       3.000
B5.BC        5.000       5.000       5.000       2.000
B6.BA        2.000       2.000       2.000       5.000
B6.BC        3.000       3.000       3.000       5.000
B7.BA        1.000       1.000       1.000       4.000
B7.BC        2.000       2.000       2.000       4.000
C1.CA          .           .           .         3.000
C1.CB        2.000       2.000       2.000       5.000
C2.CA        1.000       1.000       1.000       2.000
C2.CB        3.000       3.000       3.000       4.000
C3.CA        2.000       2.000       2.000       1.000
C3.CB        5.000       5.000       5.000       3.000
C4.CA        3.000       3.000       3.000       1.000
C4.CB        8.000       8.000       8.000       2.000
C5.CA        2.000       2.000       2.000       3.000
C5.CB        5.000       5.000       5.000       1.000
C6.CA          .           .           .         2.000
C6.CB        3.000       3.000       3.000       1.000
```

C7.CA	.	.	.	1.000
C7.CB	2.000	2.000	2.000	2.000

Appendix V: Results for Example Problem (Version 5)

This appendix contains a portion of the results from the GAMS program for the fifth version of the example problem in Chapter IV.

**** OBJECTIVE VALUE 278.0000

EQU SUPPLY CONSERVATION OF FLOW FOR SUPPLY NODES

	LOWER	LEVEL	UPPER	MARGINAL
A2.AC	2.000	2.000	2.000	2.000
A3.AB	6.000	6.000	6.000	2.000
A3.AC	2.000	2.000	2.000	1.000
A4.AB	12.000	12.000	12.000	2.000
A4.AC	5.000	5.000	5.000	3.000
A5.AB	6.000	6.000	6.000	1.000
A5.AC	3.000	3.000	3.000	2.000
A6.AB	5.000	5.000	5.000	4.000
A6.AC	2.000	2.000	2.000	5.000
A7.AB	2.000	2.000	2.000	3.000
A7.AC	1.000	1.000	1.000	4.000
B1.BA	1.000	1.000	1.000	2.000
B1.BC	2.000	2.000	2.000	1.000
B2.BA	2.000	2.000	2.000	3.000
B2.BC	3.000	3.000	3.000	2.000
B3.BA	2.000	2.000	2.000	2.000
B3.BC	5.000	5.000	5.000	1.000
B4.BA	4.000	4.000	4.000	5.000
B4.BC	8.000	8.000	8.000	1.000
B5.BA	2.000	2.000	2.000	4.000
B5.BC	5.000	5.000	5.000	2.000
B6.BA	2.000	2.000	2.000	3.000
B6.BC	3.000	3.000	3.000	1.000
B7.BA	1.000	1.000	1.000	3.000
B7.BC	2.000	2.000	2.000	2.000
C1.CA	.	.	.	1.000
C1.CB	2.000	2.000	2.000	2.000
C2.CA	1.000	1.000	1.000	1.000
C2.CB	3.000	3.000	3.000	3.000
C3.CA	2.000	2.000	2.000	2.000
C3.CB	5.000	5.000	5.000	2.000
C4.CA	3.000	3.000	3.000	1.000
C4.CB	8.000	8.000	8.000	1.000
C5.CA	2.000	2.000	2.000	4.000
C5.CB	5.000	5.000	5.000	1.000
C6.CA	.	.	.	3.000
C6.CB	3.000	3.000	3.000	4.000

C7.CA	.	.	.	2.000
C7.CB	2.000	.000	2.000	3.000

Appendix W: GAMS Program for Example Problem (Version 6)

This appendix contains the GAMS program for the sixth version of the example problem in Chapter IV.

```
SET K   commodities (cargo)
 /AB, AC, BA, BC, CA, CB/;

SET I   airbases(AB)-time periods
 /A1 * A7, B1 * B7, C1 * C7, D8/;

ALIAS (I,J);

ALIAS (I,IP);

SET DIK(I,K)  dynamic set for IK;
DIK(I,K) = yes;

SET E1(I,J,K)  arcs for cargo staying at AB
 /(A1.A2, A2.A3, A3.A4, A4.A5, A5.A6, A6.A7, A7.A1,
  B1.B2, B2.B3, B3.B4, B4.B5, B5.B6, B6.B7, B7.B1,
  C1.C2, C2.C3, C3.C4, C4.C5, C5.C6, C6.C7, C7.C1).
  (AB, AC, BA, BC, CA, CB)/;

SET Et(I,J,K)  dynamic set for E1;
Et(I,J,K) = no;
Et(E1) = yes;

SET E2(I,J,K)  arcs representing a-c with cargo
 /(C1.A2, A2.B3, B3.C4, C4.A5, A5.B6, B6.C7,
  B4.D8, D8.C5, C5.A6, A6.C7, C7.B1,
  B4.C5, C5.B6).
  (AB, AC, BA, BC, CA, CB)/;

SET Es(I,J,K)  dynamic set for E2;
Es(I,J,K) = no;
Es(E2) = yes;

SET E(I,J,K)  set of all arcs (Et and Es);
E(I,J,K) = Et(I,J,K) + Es(I,J,K);

SET E3(I,J)  arcs representing aircraft
 /C1 .A2, A2 .B3, B3.C4, C4.A5, A5.B6, B6.C7,
  B4.D8, D8.C5, C5.A6, A6.C7, C7.B1,
  B4.C5, C5.B6/

SET SIKN(I,K)  supply nodes for all cargo
 /A1.AB, A2.AB, A3.AB, A4.AB, A5.AB, A6.AB, A7.AB,
  A1.AC, A2.AC, A3.AC, A4.AC, A5.AC, A6.AC, A7.AC,
```

```
  B1.BA, B2.BA, B3.BA, B4.BA, B5.BA, B6.BA, B7.BA,
  B1.BC, B2.BC, B3.BC, B4.BC, B5.BC, B6.BC, B7.BC,
  C1.CA, C2.CA, C3.CA, C4.CA, C5.CA, C6.CA, C7.CA,
  C1.CB, C2.CB, C3.CB, C4.CB, C5.CB, C6.CB, C7.CB/;

SET SUPNODE(I,K)  dynamic set for SIKN;
SUPNODE(I,K) = no;
SUPNODE(SIKN) = yes;

SET DIKN(I,K)  airbase demand nodes for all cargo
 /A1.BA, A2.BA, A3.BA, A4.BA, A5.BA, A6.BA, A7.BA,
  A1.CA, A2.CA, A3.CA, A4.CA, A5.CA, A6.CA, A7.CA,
  B1.AB, B2.AB, B3.AB, B4.AB, B5.AB, B6.AB, B7.AB,
  B1.CB, B2.CB, B3.CB, B4.CB, B5.CB, B6.CB, B7.CB,
  C1.AC, C2.AC, C3.AC, C4.AC, C5.AC, C6.AC, C7.AC,
  C1.BC, C2.BC, C3.BC, C4.BC, C5.BC, C6.BC, C7.BC/;

SET DMDNODE(I,K)  dynamic set for DIKN;
DMDNODE(I,K) = no;
DMDNODE(DIKN) = yes;

SET ZIKN(I,K)  neither demand nor supply nodes;
ZIKN(I,K) = DIK(I,K) - SUPNODE(I,K) - DMDNODE(I,K);

SET ZN(I)  airbases that serve as zero balance nodes
 /A1 * A7, B1 * B7, C1 * C7/;

PARAMETER C(I,J,K)  delay;

C(I,J,K) = 0;

C(I,J,K)$Et(I,J,K) = 1;

C(I,J,K)$Es(I,J,K) = 1;

C("D8","C5",K) = 0

PARAMETER S(I,K) the supply at node SIKN
 /A1.AB 2, A2.AB 5, A3.AB 6, A4.AB 12, A5.AB 6, A6.AB 5,
A7.AB 2,
  A1.AC 1, A2.AC 2, A3.AC 2, A4.AC 5, A5.AC 3, A6.AC 2,
A7.AC 1,
  B1.BA 1, B2.BA 2, B3.BA 2, B4.BA 4, B5.BA 2, B6.BA 2,
B7.BA 1,
  B1.BC 2, B2.BC 3, B3.BC 5, B4.BC 8, B5.BC 5, B6.BC 3,
B7.BC 2,
  C1.CA 0, C2.CA 1, C3.CA 2, C4.CA 3, C5.CA 2, C6.CA 0,
C7.CA 0,
```

```
  C1.CB 2, C2.CB 3, C3.CB 5, C4.CB 8, C5.CB 5, C6.CB 3,
C7.CB 2/;

PARAMETER CAP(I,J)  aircraft capacity
 /C1.A2 30, A2.B3 30, B3.C4 30, C4.A5 30, A5.B6 30, B6.C7
30,
  B4.D8 25, D8.C5 25, C5.A6 25, A6.C7 25, C7.B1 25,
  B4.C5 18, C5.B6 18/;

VARIABLE
Z  total delay

POSITIVE VARIABLES
X(I,J,K) shipment quantity
SUP(K)  total supply for each cargo K
DEL(K)  total amount delivered for each cargo
UNDEL(K)  total amount not delivered for each cargo;

EQUATIONS
DELAY  objective function
SUMS(K)  total supply for each cargo K
SUPPLY(IP,K)  conservation of flow for supply nodes
DEMAND(IP,K)  conservation of flow for demand nodes
DELIVER(K)  total amount delivered for each cargo
UNDELIVER(K)  total amount not delivered for each cargo
BAL(IP,K)  conservation of flow for ZIKN nodes
UB(I,J)  upper bound capacity constraint for aircraft;

DELAY .. Z =E= SUM((I,J,K)$E(I,J,K), C(I,J,K)*X(I,J,K));

SUMS(K) .. SUP(K) =E= SUM(I,S(I,K));

SUPPLY(IP,K)$SIKN(IP,K) .. SUM(J, X(IP,J,K)$E(IP,J,K)) -
                           SUM(I, X(I,IP,K)$E(I,IP,K))
                           =E= S(IP,K);

DEMAND(IP,K)$DIKN(IP,K) .. SUM(J, X(IP,J,K)$E(IP,J,K)) -
                           SUM(I, X(I,IP,K)$E(I,IP,K))
                           =G= -SUP(K);

DELIVER(K) .. DEL(K) =E= SUM((I,IP)$E3(I,IP),
X(I,IP,K)$DIKN(IP,K));

UNDELIVER(K) .. UNDEL(K) =E= SUP(K) - DEL(K);

BAL(IP,K)$ZIKN(IP,K) ..  SUM(J, X(IP,J,K)$E(IP,J,K)) -
                         SUM(I, X(I,IP,K)$E(I,IP,K))
                         =E= 0;

UB( E3(I,J) ) ..  SUM(K, X(I,J,K)) =L= CAP(E3);
```

```
MODEL MMCF /ALL/;

OPTION ITERLIM = 10000, RESLIM = 10000;
OPTION LIMROW = 0, LIMCOL = 0;

SOLVE MMCF USING LP MINIMIZING Z;
```

Appendix X: Results for Example Problem (Version 6)

This appendix contains a portion of the results from the GAMS program for the sixth version of the example problem in Chapter IV.

```
**** OBJECTIVE VALUE              274.0000

     EQU SUPPLY      CONSERVATION OF FLOW FOR SUPPLY NODES

             LOWER       LEVEL       UPPER     MARGINAL

A2.AC        2.000       2.000       2.000       2.000
A3.AB        6.000       6.000       6.000       3.000
A3.AC        2.000       2.000       2.000       4.000
A4.AB       12.000      12.000      12.000       2.000
A4.AC        5.000       5.000       5.000       3.000
A5.AB        6.000       6.000       6.000       1.000
A5.AC        3.000       3.000       3.000       2.000
A6.AB        5.000       5.000       5.000       2.000
A6.AC        2.000       2.000       2.000       1.000
A7.AB        2.000       2.000       2.000       3.000
A7.AC        1.000       1.000       1.000       4.000
B1.BA        1.000       1.000       1.000       4.000
B1.BC        2.000       2.000       2.000       3.000
B2.BA        2.000       2.000       2.000       3.000
B2.BC        3.000       3.000       3.000       2.000
B3.BA        2.000       2.000       2.000       2.000
B3.BC        5.000       5.000       5.000       1.000
B4.BA        4.000       4.000       4.000       2.000
B4.BC        8.000       8.000       8.000       1.000
B5.BA        2.000       2.000       2.000       4.000
B5.BC        5.000       5.000       5.000       2.000
B6.BA        2.000       2.000       2.000       3.000
B6.BC        3.000       3.000       3.000       1.000
B7.BA        1.000       1.000       1.000       5.000
B7.BC        2.000       2.000       2.000       4.000
C1.CA          .           .           .        -2.000
C1.CB        2.000       2.000       2.000       2.000
C2.CA        1.000       1.000       1.000       3.000
C2.CB        3.000       3.000       3.000       4.000
C3.CA        2.000       2.000       2.000       2.000
C3.CB        5.000       5.000       5.000       3.000
C4.CA        3.000       3.000       3.000       1.000
C4.CB        8.000       8.000       8.000       2.000
C5.CA        2.000       2.000       2.000       1.000
C5.CB        5.000       5.000       5.000       1.000
C6.CA          .           .           .          EPS
C6.CB        3.000       3.000       3.000       2.000
```

C7.CA	.	.	.	-1.000
C7.CB	2.000	2.000	2.000	1.000

Appendix Y: Results for Example Problem (Version 7)

This appendix contains a portion of the results from the GAMS program for the seventh version of the example problem in Chapter IV.

```
**** OBJECTIVE VALUE              308.0000

     EQU SUPPLY      CONSERVATION OF FLOW FOR SUPPLY NODES

           LOWER      LEVEL      UPPER    MARGINAL

A2.AC      2.000      2.000      2.000      2.000
A3.AB      6.000      6.000      6.000      5.000
A3.AC      2.000      2.000      2.000      4.000
A4.AB     12.000     12.000     12.000      4.000
A4.AC      5.000      5.000      5.000      3.000
A5.AB      6.000      6.000      6.000      3.000
A5.AC      3.000      3.000      3.000      2.000
A6.AB      5.000      5.000      5.000      2.000
A6.AC      2.000      2.000      2.000      1.000
A7.AB      2.000      2.000      2.000      3.000
A7.AC      1.000      1.000      1.000      4.000
B1.BA      1.000      1.000      1.000      5.000
B1.BC      2.000      2.000      2.000      3.000
B2.BA      2.000      2.000      2.000      4.000
B2.BC      3.000      3.000      3.000      2.000
B3.BA      2.000      2.000      2.000      3.000
B3.BC      5.000      5.000      5.000      1.000
B4.BA      4.000      4.000      4.000      2.000
B4.BC      8.000      8.000      8.000      1.000
B5.BA      2.000      2.000      2.000      4.000
B5.BC      5.000      5.000      5.000      3.000
B6.BA      2.000      2.000      2.000      3.000
B6.BC      3.000      3.000      3.000      2.000
B7.BA      1.000      1.000      1.000      2.000
B7.BC      2.000      2.000      2.000      1.000
C1.CA        .          .          .        1.000
C1.CB      2.000      2.000      2.000      2.000
C2.CA      1.000      1.000      1.000      4.000
C2.CB      3.000      3.000      3.000      6.000
C3.CA      2.000      2.000      2.000      3.000
C3.CB      5.000      5.000      5.000      5.000
C4.CA      3.000      3.000      3.000      2.000
C4.CB      8.000      8.000      8.000      4.000
C5.CA      2.000      2.000      2.000      1.000
C5.CB      5.000      5.000      5.000      3.000
C6.CA        .          .          .         EPS
C6.CB      3.000      3.000      3.000      2.000
```

C7.CA	.	.	.	-1.000
C7.CB	2.000	2.000	2.000	1.000

Bibliography

Ackley, M., W. Carter, G. Hughes, J. Litko, K. Ware, A. Whisman, and R. Roehrkasse. *Optimization Applications at the Military Airlift Command: Importance and Difficulties.* Unpublished paper provided to the Second International Conference on Industrial and Applied Mathematics. Washington, D.C., July 8-12, 1991.

Ackley, M., B. Carter, J. Litko, K. Ware and A. Whisman. *Decision Support for the Air Mobility Command Scheduled Cargo System.* Unpublished paper provided by LTC J. Litko, Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL.

Ali I., D. Barnett, K. Farhangian, J. Kennington, B. McCarl, B. Patty, B. Shettey and P. Wong. "Multicommodity Network Problems: Applications and Computations," *IIE Transactions,* 16: 127-134 (June 1984).

Bellmore, M., G. Bennington and S. Lubore. "A Multivehicle Tanker Scheduling Problem," *Transportation Science,* 5: 36-47 (June 1971).

Bodin, Lawrence D. "Twenty Years of Routing and Scheduling," *Operations Research,* 38: 571-579 (July-August 1990).

Borsi, MAJ John. Personal interview. Air Force Institute of Technology, Wright-Patterson AFB, OH. 6 August 1992.

Borsi, MAJ John. Personal interview. Air Force Institute of Technology, Wright-Patterson AFB, OH. 28 August 1992.

Borsi, MAJ John. Personal interview. Air Force Institute of Technology, Wright-Patterson AFB, OH. 8 February 1993.

Carter, Brand and Joseph R. Litko. *Simulating the Air Mobility Command Channel Cargo System.* Unpublished paper provided by LTC J. Litko, Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL.

Clarke, S. and J. Surkis. "An Operations Research Approach to Racial Desegregation of School Systems," *Socio-Economic Planning Sciences,* 1: 259-272 (January 1968).

Gertsbakh, Ilya and Paolo Serafini. "Periodic Transportation Schedules with Flexible Departure Times," *European Journal of Operational Research*, 50: 298-309 (February 1991).

Hanson, MAJ Reed. Telephone interview. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 9 September 1992.

Helgason, R.V. and J.L. Kennington. "A Product Form Representation of the Inverse of a Multicommodity Cycle Matrix," *Networks*, 7: 297-322 (1977).

Kennington, Jeff L. "A Survey of Linear Cost Multicommodity Network Flows," *Operations Research*, 26: 209-236 (March-April 1978).

Kikuchi, Shinya and Jong-Ho Rhee. "Scheduling Method for Demand-Responsive Transportation System," *Journal of Transportation Engineering*, 115: 630-645 (November 1989).

Litko, LTC J. Personal interview. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 22 September 1992.

Litko, LTC J. Telephone interview. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 26 August 1992.

Litko, LTC J. Telephone interview. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 9 September 1992.

Litko, LTC J. Telephone interview. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 13 October 1992.

Moul, Capt Justin E. *A Method for Determining Schedule Delay Information in a Channel Cargo Route Network Schedule*. MS thesis, AFIT/GST/ENS/92M-05. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, June 1992 (Note: No DTIC assigned yet).

Rau, CPT Gregory S. *Scheduling Air Mobility Command's Channel Cargo Missions*. MS thesis, AFIT/GOR/ENS/93M-19. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1993 (Note: No DTIC assigned yet).

Robinson, 1LT J. Personal interview and data derived from an AMC study. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 22 September 1992.

Solanki, Rajendra S. and Frank Southworth. "An Execution Planning Algorithm for Military Airlift," *Interfaces*, 21: 121-131 (July-August 1991).

Whisman, Alan. Correspondence. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 27 October 1992.

Whisman, Alan. Personal interview. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 22 September 1992.
Whisman, Alan. Telephone interview. Command Analysis Group, Air Mobility Command, HQ AMC/XPYR, Scott AFB, IL. 30 October 1992.

White W.W. and E. Wrathall. *A System for Railroad Traffic Scheduling*. Technical Report No. 320-2993. IBM Corporation, Philadelphia Scientific Center, Philadelphia, Pennsylvania, August 1970.

Wollmer, R.D. "Multicommodity Networks with Resource Constraints: The Generalized Multicommodity Flow Problem," *Networks*, 1: 245-263 (1972).

Zanakis, Stelios H., James R. Evans, and Alkis A. Vazacopoulos. "Heuristic Methods and Applications: A Categorized Survey," *European Journal of Operational Research*, 43: 88-110 (November 1989).

Vita

Captain Michael Del Rosario was born on 1 December 1960 in El Paso, Texas. He graduated from Irvin High School in 1979 and attended the United States Military Academy, graduating with a Bachelor of Science (specialty: Civil Engineering) in May 1983. Upon graduation, he attended the Engineer Officer's Basic Course and was assigned to the 52nd Engineer Battalion at Fort Carson, Colorado, where he served as platoon leader, company executive officer, and battalion construction engineer. After graduating from the Engineer Officer's Advance Course in June 1987, Captain Del Rosario was assigned to the 249th Engineer Battalion in Karlsruhe, Germany, where he served as battalion S-2/warplans officer, battalion assistant S-3/civil engineer, and company commander of C Company. Captain Del Rosario also commanded C Company from December 1990 to March 1991 during the battalions's deployment to Southwest Asia for Operations Desert Shield and Desert Storm. After his overseas tour, he entered the School of Engineering, Air Force Institute of Technology in August 1991. Captain Del Rosario is a registered Professional Engineer in the state of Virginia. He and his wife, Teresa, have three children: Gregory, Michelle, and David.

Permanent Address: 5232 Marie Tobin
El Paso, Texas 79924